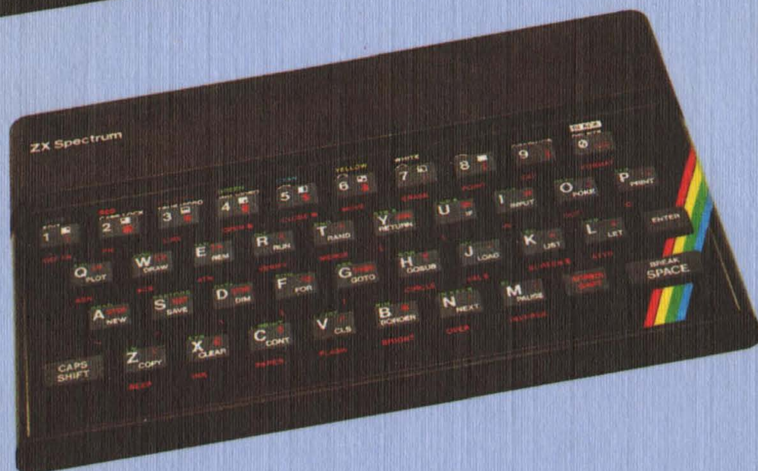
HARDWARE-ERWEITERUNG FÜR ZX SPECTRUM



Jörg Reinmuth



Jörg Reinmuth

Hardware-Erweiterungen für ZX Spectrum

Brandenburgisches
Verlagshaus

Reinmuth, J.:
Hardware-Erweiterungen für ZX Spectrum. –
Berlin: Brandenburgisches Verlagshaus, 1990. –
96 Seiten: 87 Bilder, 19 Tabellen, 5 Programme

ISBN 3-327-01030-7

1. Auflage

Computersatz: System SCLX
Druck und buchbinderische Weiterverarbeitung:
Druckhaus Schöneweide – 3 4380-0
Lektor: Steffen Würtenberger
Zeichnungen: Berit Peschel
Umschlaggestaltung: Rosemarie Lebek
Redaktionsschluß: 15. Januar 1990
LSV 3539
Bestellnummer: 747 3841
DDR 6,20 M

Vorwort

Die Vorzüge von Mikroelektronik und Computertechnik sind sicher jedem Leser ausreichend bekannt. In den letzten Jahren hat der Computer auch den Heim-, Hobby- und Amateurbereich erobert. In der DDR dominieren zumeist 8-bit-Systeme, die überwiegend mit der CPU *U 880* arbeiten. Dieser zum *Z 80* kompatible Mikroprozessor kann aufgrund seiner weiten, auch internationalen Verbreitung und seiner Leistungsfähigkeit schlechthin als **der** 8-bit-Prozessor bezeichnet werden.
Vertreter dieser Geräteklasse aus DDR-Produktion sind z. B. *KC 85*, *Z 9001*, *Z 1013* oder *AC 1*. Als internationaler Vertreter sei hier der *ZX Spectrum* genannt, der inzwischen als Original oder durch einige mehr oder weniger kompatible Nachbauten in der DDR sowie im gesamten RGW-Bereich eine so weite Verbreitung gefunden hat, daß es Verlag und Autor angemessen erschien, diese Broschüre herauszugeben.
Das vorliegende Heft wendet sich an den fortgeschrittenen Amateur, der mit einfachen und überschaubaren Schaltungen die Leistungsfähigkeit seines Heimcomputers erhöhen möchte. Einige Kenntnisse der digitalen Schaltungstechnik werden dabei ebenso vorausgesetzt wie etwas Lötpraxis und Erfahrung im Umgang mit elektronischen Bauelementen. Die vorgestellten Schaltungen sind, eventuell mit kleinen Änderungen, an andere *U-880*-Systeme anpaßbar, obwohl in dieser Broschüre immer auf den *ZX Spectrum* Bezug genommen wird. Neben einer detaillierten Beschreibung der Hardware wird auch Treibersoftware angegeben, um die Inbetriebnahme der Schaltungen zu erleichtern.

Leipzig, im Frühjahr 1989

Jörg Reinmuth

1. Das *U-880*-Prozessorsystem

Das Mikroprozessorsystem *U 880* stellt eine Weiterentwicklung der Systeme *8080* und *8085* dar. Zum System gehören folgende LSI-Schaltkreise:

- *U 880*, CPU (zentrale Verarbeitungseinheit);
- *U 855*, PIO (parallele Ein-/Ausgabe);
- *U 856*, SIO (serielle Ein-/Ausgabe);
- *U 857*, CTC (Zähler/Zeitgeber);
- *U 858*, DMA (direkter Speicherzugriff).

Alle Schaltkreise werden vom VEB Mikroelektronik "Karl Marx" Erfurt in n-Kanal-Silicon-Gate-Technologie hergestellt und mit Ausnahme des CTC in einem 40-poligen Plastgehäuse verkappt. Der CTC wird in einem 28poligen Gehäuse geliefert. Die Architektur der Schaltkreise realisiert ein einheitliches System mit folgenden Merkmalen [1]:

- Signalzuführung über zwei Systembusse (Adreß- und Datenbus) und eine Reihe von Steuerleitungen (Steuerbus);
- einheitliches Interruptregime mit der Möglichkeit der Prioritätsverkettung der Peripherieschaltkreise;
- hochgradige Softwareprogrammierbarkeit der schaltkreisspezifischen Eigenschaften;
- gemeinsamer Einphasensystemtakt;
- einheitliche Spannungsversorgung von +5 V.

Das System wird komplettiert durch den Einsatz von Festwertspeichern, Schreib-Lese-Speichern und Standard-Logikschaltkreisen.

Neuere Interface-Schaltkreise wie die Typen *U 82530* (SCC: Serial Communication Controller - serielle Ein-/Ausgabesteuerung) und *U 82536* (CIO: Counter-/Timer and parallel I/O unit - Zähler/Zeitgeber und parallele Ein-/Ausgabeeinheit) werden hier nicht besprochen, da sie zum Zeitpunkt der Manuskripterarbeitung für den Amateur noch nicht greifbar waren. Sie werden aber in Zukunft für komplexe Systemlösungen zunehmende Bedeutung erlangen.

1.1. CPU *U 880*

Der Prozessor *U 880* ist das Herz des Systems. Er ist aufwärtskompatibel zum weit verbreiteten Prozessor *8080*, d. h. alle im *8080* enthaltenen Möglichkeiten sind auch im *U 880* vorhanden.

Die Weiterentwicklung der *U-880*-CPU (CPU: Central Processing Unit - zentrale Verarbeitungseinheit) bezieht sich vor allem auf einen effizienteren Befehlssatz und auf das Interruptverhalten.

Der Befehlssatz des *U 880* umfaßt 158 Grundbefehle mit 16-bit-, 8-bit-, 4-bit- und Einzelbitoperationen sowie Ein-/Ausgabe- und Blockbefehlen. Der Schaltkreis enthält einen internen statischen RAM, zu dem der Programmierer Zugriff hat. Dieser RAM ist unterteilt in 18 8-bit-Re-

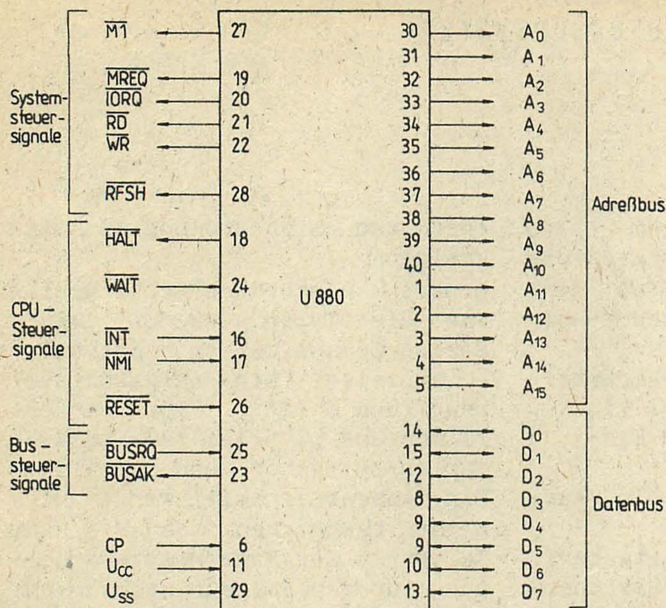


Bild 1.1 Schematische Anschlußbelegung der CPU U 880

gister und 4 16-bit-Register. Die Anschlußbelegung der CPU *U 880* zeigt Bild 1.1. Über Befehlssatz und Programmierung geben z. B. [1], [2] und [3] umfassend Auskunft.

1.2. PIO *U 855*

Die PIO (Parallel Input/Output - parallele Ein-/Ausgabe) *U 855* ist die parallele Ein-/Ausgabeeinheit des Systems *U 880.*

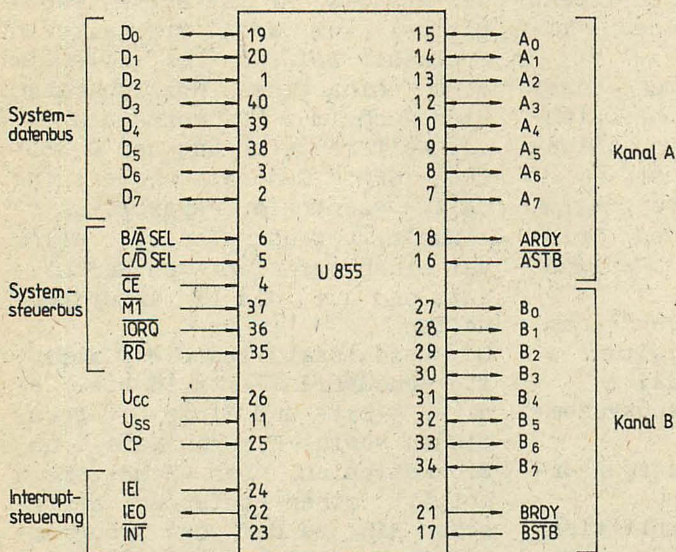


Bild 1.2 Schematische Anschlußbelegung der PIO U 855

Der Schaltkreis enthält zwei komplette, nach außen TTL-kompatible 8-bit-Ein-/Ausgabetore einschließlich der dazugehörigen Steuerlogik und ist in seinen wesentlichen Eigenschaften softwareprogrammierbar. Bild 1.2 zeigt die Anschlußbelegung. Die Zusammenschaltung der PIO *U 855* mit dem Prozessor *U 880* erfordert keine weiteren Standardbauelemente, ausgenommen Adreßdekoder, Datenbustreiber und Interrupt-Kaskadierungslogik bei den größeren Systemen. In Abhängigkeit von der Taktfrequenz des Prozessors kommen die PIO-Typen *UA 855* (4 MHz), *UB 855* (2,5 MHz) und *UB 855 S1* (1 MHz) zum Einsatz. Im allgemeinen lassen sich die Schaltkreise aber auch bei höheren als den angegebenen Frequenzen betreiben. Die PIO belegt im Adreßraum eines Mikrorechners 4 Adressen, die paarweise den Kanälen A und B als Steuer- bzw. Datenwortadressen zugeordnet sind. Unter Verwendung der PIO *U 855* kann der gesamte Datentransfer eines Mikrorechners unter vollständiger Interruptüberwachung durchgeführt werden. Diese Problematik wird hier nicht ausführlich behandelt. Weil die PIO *U 855* aber Bestandteil einiger Schaltungen dieses Buches ist, wird im folgenden ihre Programmierung kurz dargestellt. Weiterführende Einzelheiten können z. B. [1], [2] entnommen werden. Die Programmierung der PIO sowie aller anderen Peripherieschaltkreise erfolgt mit OUT-Befehlen an die entsprechende Steuerwortadresse. Die PIO ist für den Betrieb mit der CPU *U 880* in der sehr leistungsfähigen Interruptbetriebsart 2 ausgelegt. In dieser Betriebsart wird innerhalb eines Interruptanerkennungszyklus der CPU von der PIO ein sogenannter Interruptvektor auf den Datenbus gelegt, der (als niederwertiger Teil) zusammen mit dem Inhalt des I-Registers der CPU (als höherwertiger Teil) einen 16-bit-Zeiger bildet, der auf eine Speicheradresse zeigt. Dieser und der folgende Speicherplatz enthält die Startadresse für die Interruptroutine. Der Interruptvektor wird der PIO als erstes Steuerwort mitgeteilt; er kann entfallen, wenn in keiner Interruptbetriebsart gearbeitet wird. Das Steuerwort hat folgendes Format:

D7	D6	D5	D4	D3	D2	D1	D0
V7	V6	V5	V4	V3	V2	V1	V0

Mit PIO-Kanal A sind 4 Betriebsarten (Moden) realisierbar:

Mode 0: Byte-Ausgabe
Mode 1: Byte-Eingabe
Mode 2: Byte-Ein-/Ausgabe
Mode 3: Bit-Ein-/Ausgabe

Kanal B kann nur in den Betriebsarten 0, 1 und 3 arbeiten. Das Betriebsartensteuerwort hat das Format

D7	D6	D5	D4	D3	D2	D1	D0
M1	M0	x	x	1	1	1	1

mit folgender Zuordnung von M1 und M0 zu den Betriebsarten:

M1	M0	MODE
0	0	0
0	1	1
1	0	2
1	1	3

Nicht verwendete Bits werden hier und im folgenden mit x gekennzeichnet und zur Vereinheitlichung konsequent auf 0 gesetzt. Nur für die Betriebsart 3 (in der alle acht Portleitungen eines Kanals unabhängig vonein-

ander als Ein- oder Ausgänge definiert werden können) gelten 3 weitere Steuerwörter. Mit dem ersten wird die Funktion der Portleitungen als Ein- oder Ausgang festgelegt. Dabei gilt die Zuordnung 0 = Ausgang, 1 = Eingang.

D7	D6	D5	D4
E/A7	E/A6	E/A5	E/A4

D3	D2	D1	D0
E/A3	E/A2	E/A1	E/A0

Das nächste Steuerwort dient der Interruptsteuerung. Es kann wie auch das folgende wieder entfallen, wenn in keiner Interruptbetriebsart gearbeitet wird.

D7	D6	D5	D4
INT	UND/ ODER	LOW/ HIGH	Maske folgt

D3	D2	D1	D0
0	1	1	1

D7 = 0: Interrupt gesperrt
D7 = 1: Interrupt freigegeben
D6 = 0: UND-Verknüpfung der interruptauslösenden Leitungen
D6 = 1: ODER-Verknüpfung der interruptauslösenden Leitungen
D5 = 0: aktiver Pegel LOW
D5 = 1: aktiver Pegel HIGH
D4 = 0: Maske folgt nicht
D4 = 1: Maske folgt

Sofern im Interruptsteuerwort bit 4 auf 1 gesetzt wurde, wertet der PIO-Kanal das nächste an ihn gerichtete Steuerwort als Maske. Damit wird festgelegt, welche der Kanalleitungen auf die Interruptbildung Einfluß nehmen. Die Kanalleitungen haben Einfluß auf die Interruptbildung, wenn das entsprechende Bit im Maskierungssteuerwort auf 0 gesetzt ist.

D7	D6	D5	D4
MB7	MB6	MB5	MB4

D3	D2	D1	D0
MB3	MB2	MB1	MB0

Das letzte Steuerwort hat wieder Bedeutung für alle 4 Betriebsarten der PIO. Es hat das Format

D7	D6	D5	D4
Interrupt-Freigabe	x	x	x

D3	D2	D1	D0
0	0	1	1

und dient der Beeinflussung der Interruptfreigabe. Mit D7 = 0 wird die Weiterleitung eines Interrupts verhindert, mit D7 = 1 erlaubt. Auch wenn nicht in einer Interruptbetriebsart gearbeitet wird, sollte dieses Steuerwort (dann mit D7 = 0) ausgegeben werden, um unvorhersehbare Interruptanforderungen mit Sicherheit auszuschließen.

1.3. SIO *U 856*

Die SIO *U 856* ist die zum System *U 880* gehörige serielle Ein-/Ausgabe-Einheit (SIO: Serial Input/Output - serielle Ein-/Ausgabe). Der Schaltkreis enthält 4 unabhängige serielle Ports, die in je 2 Sender- und Empfängerports unterteilt sind. Durch Software programmierbar sind eine Vielzahl von Betriebsarten. Einstellbar sind unter anderem die Anzahl der übertragenen Bits pro Zeichen, Anzahl der Stopbits, Art der Parität (gerade, ungerade oder keine), verschiedene Taktvarianten, automatische Prüfsummenbildung und -kontrolle, Null- und Flag-Einfügung und vieles andere mehr. Details zur umfangreichen

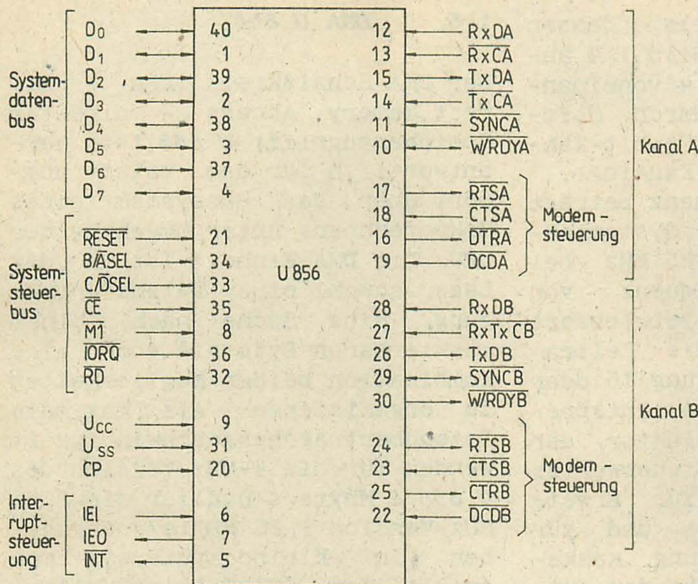


Bild 1.3 Schematische Anschlußbelegung der SIO U 856

Programmierung der SIO *U 856* können [2] entnommen werden. Die Anschlußbelegung zeigt Bild 1.3.

1.4. CTC *U 857*

Der CTC *U 857* ist der Zähler-/Zeitgeber-Baustein des Systems U 880 (CTC: Counter/Timer-Circuit - Zähler/Zeitgeber-Schalt-

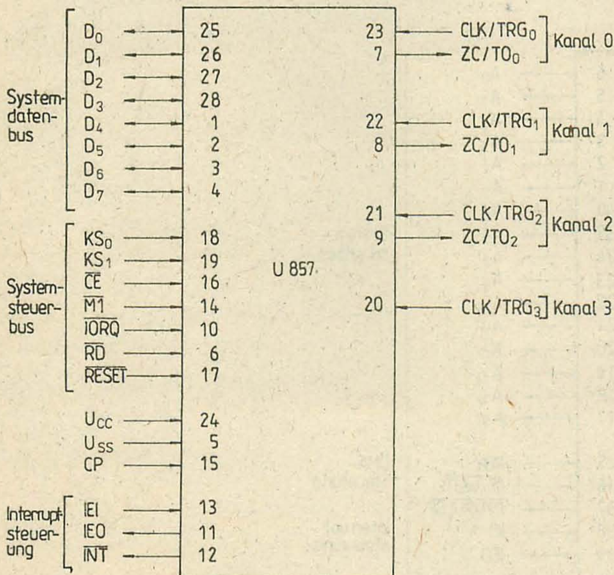


Bild 1.4 Schematische Anschlußbelegung des CTC U 857

Kreis). Der Schaltkreis, dessen Anschlußbelegung im Bild 1.4 angegeben ist, enthält 4 voneinander unabhängige und durch Software programmierbare 8-bit-Zähler/16-bit-Zeitgeber-Kanäle.
Die maximale Zälfrequenz beträgt stets die Hälfte des Systemtaktes, also z. B. 1,25 MHz bei einer Systemtaktfrequenz von 2,5 MHz. In der Betriebsart Zeitgeber beträgt die Zeitraster-Intervall-Auflösung 16 oder 256 Systemtaktperioden entsprechend einem Vorteilerfaktor, der Bestandteil eines Steuerwortes ist. Die Kanäle sind zur Erweiterung des Zählumfangs und zur gegenseitigen Triggerung kaskadierbar. Allerdings ist der Ausgang des Kanals 3 wegen der Beschränkung auf ein 28poliges Gehäuse nicht herausgeführt. Über die Programmierung geben [1], [2] Auskunft.

1.5. DMA *U 858*

Der DMA-Schaltkreis (DMA - Direct Memory Access - direkter Speicherzugriff) *U 858* ist verantwortlich für den Datentransport über das Bussystem eines Mikrorechners unter Umgehung der CPU. Die DMA-Einheit ist in der Lage, sowohl eine Datenübertragung, eine Suche nach einem maskierbaren Byte als auch eine Kombination beider Möglichkeiten zu organisieren. Als maximale Datenübertragungsgeschwindigkeit werden für die 4-MHz-Version des *U 858* 2 MByte/s und für die 2,5-MHz-Version 1,25 MByte/s angegeben. In Kleincomputersystemen spielt der DMA-Betrieb allerdings nur eine untergeordnete Rolle, zumal im Befehlssatz der CPU *U 880* sehr leistungsfähige Blocktransfer- (z. B. LDIR, OTIR, INIR) und Blocksuchbefehle (z. B. CPIR) zur Verfügung stehen. Bild 1.5 zeigt die Anschlußbelegung des *U 858*.

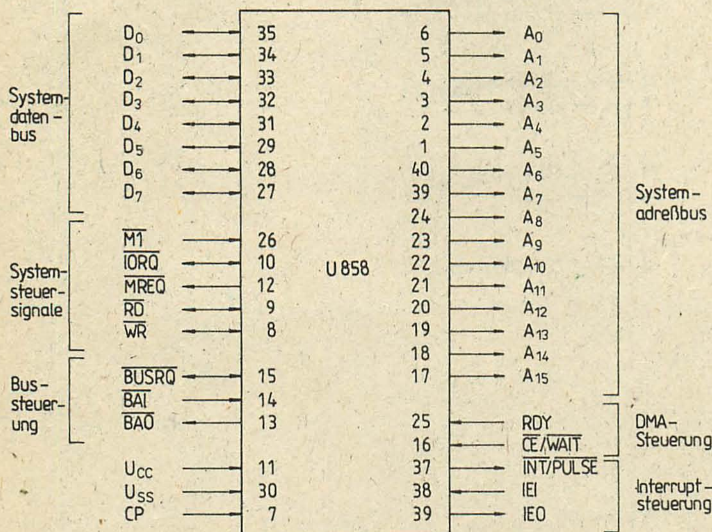


Bild 1.5 Schematische Anschlußbelegung des DMA U 858

2. Der Aufbau eines Kleincomputers

Der Aufbau aller Kleincomputer ist im Prinzip ähnlich. Sie enthalten, meist auf nur einer Platine, einen Mikroprozessor, Speicher und verschiedene Ein-/Ausgabe-Einheiten sowie Schaltungen zur Betriebsspannungsversorgung. Daraus ergibt sich das einfache Übersichtsschaltbild nach Bild 2.1. Der Mikroprozessor sorgt für die Ausführung der im Speicher abgelegten Befehle und ist für die Verwaltung des gesamten Systems verantwortlich. Im Speicher sind Daten und Programme abgelegt. Es existieren verschiedene Speichertypen, die in Nur-Lese-Speicher (Read-Only Memory - ROM) und Schreib-Lese-Speicher oder Speicher mit wahlfreiem Zugriff (Random-Access-Memory - RAM) unterschieden

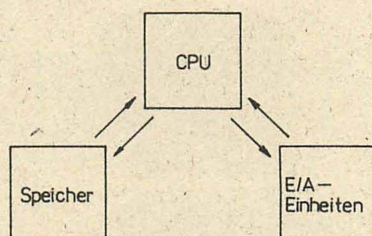


Bild 2.1 Einfaches Übersichtsschaltbild eines Kleincomputers

werden. Die Ein-/Ausgabe-Einheiten dienen hauptsächlich der Bediener-Computer-Kommunikation. In erster Linie zu nennen sind hier Tastatur und Bildschirm. Dazu gehört aber auch der Anschluß externer Massenspeicher (Magnetband oder Floppy-Disk) zur dauerhaften Speicherung von Programmen und Daten.

3. Spezialschaltkreise für Kleincomputer

Im Hinblick auf eine Minimierung der Hardware und damit verbundene geringere Produktionskosten und Ausfallwahrscheinlichkeiten existieren international eine Vielzahl von Schaltkreisen, die speziell für Kleincomputer, ja oft für einen speziellen Gerätetyp entworfen wurden. Sie sind meist mit einem Kürzel entsprechend Ihrer Funktion benannt - Beispiele dafür sind SID (Sound Generator), VIC (Video Controller) oder das aus dem *ZX Spectrum* bekannte ULA (Uncommitted Logic Array) - und ersetzen eine oft große Zahl von Standardbauelementen, denen sie durch ihre teilweise Programmierbarkeit oft noch überlegen sind. So ist das *ZX-Spectrum*-ULA z. B. verantwortlich für den Bildaufbau, die Tastaturabfrage, die Kontrolle der Ein-/Ausgabeeinheiten für Kassettengerät und Lautsprecher, ferner für die Taktaufbereitung, die Interruptgenerierung, Dekodierung verschiedener Speicherbereiche, Erzeugung spezieller Signale für die dynamischen Speicher und anderes mehr [4], [5], [6]. Diesem ULA und einem weiteren Schaltkreis, der die gesamte PAL-Aufbereitung des Video-Signals übernimmt, ist es wohl in erster Linie zu verdanken, daß (natürlich mit einem fast genial zu nennenden Systemkonzept) der *ZX Spectrum* ein Rechner in einem winzigen Gehäuse ist, der keine Konkurrenz in seiner Klasse scheuen muß. Das gilt umsomehr, wenn man ihn mit etwas Zusatzhardware zu noch höherer Leistungsfähigkeit ausgebaut hat. Als nachteilig an der Verwendung dieser speziellen Schaltkreise erweist sich, daß sie im Reparaturfall sehr schwer beschaffbar sind und daß über sie meist nur sehr spärliche technische Unterlagen existieren.

4. Adreßdekodierung

Damit ein peripheres Bauelement in einem Computersystem überhaupt angesprochen wird, ist ein sogenannter Geräteauswahlimpuls notwendig, der über einen Eingang dieses Bauelement aktiviert. Dieser Eingang wird oft mit CS (Chip Select - Bausteinauswahl) oder CE (Chip Enable - Bausteinfreigabe) bezeichnet und ist bei den Peripheriebausteinen des Systems *U 880* low-aktiv.
Zur Bildung dieses Auswahlimpulses werden i. allg. die 8 niederwertigen Adressen A0 bis A7 und die Signale $\overline{IORQ}$, $\overline{WR}$ und $\overline{RD}$ über entsprechende Dekoder oder Gatter verknüpft. Dabei wird zwischen ein- und mehrdeutiger Dekodierung unterschieden.
Bei der mehrdeutigen Dekodierung wird nur ein Teil der Adreßleitungen bzw. nur eine einzelne Adreßleitung zur Auswahl des jeweiligen peripheren Bauelementes verwendet. So spricht z. B. die Adreßleitung A0 im *ZX Spectrum* direkt das ULA an (mit A0 = 0 und $\overline{IORQ}$ = 0). Solche Maßnahmen vereinfachen die Hardware eines Computers zwar erheblich, haben aber bei der Hardware-Erweiterung eines solchen Rechners gewisse Beschränkungen zur Folge. Bei konsequenter Fortführung dieser Technik lassen sich nämlich insgesamt nur 8 E/A-Adressen (entsprechend der 8 Adreßbits) bilden, wovon eine einzige PIO schon 4 fordert.
Darüber hinaus ergeben sich aber auch softwareseitige Konsequenzen. So ist der Peripheriebaustein, der mit A0 = 0 angesprochen wird, über OUT-Befehle an alle geradzahligen Adressen (also 128!) erreichbar. Günstiger, besonders bei umfangreicheren Hardware-Erweiterungen, ist deshalb die eindeutige Dekodierung. Hierbei kann jede Adresse zwischen 0 und 255 genau ein peripheres Gerät ansprechen. Die Hardware des *ZX Spectrum* zwingt hier allerdings zu einem Kompromiß, da gesichert werden muß, daß ULA (A0) und ZX-Drucker (A1) nie gleichzeitig mit einem anderen Peripheriegerät aktiviert werden. Mit den verbleibenden sechs Adreßleitungen sind aber immer noch 2^6 = 64 E/A-Einheiten ansprechbar.

5. Hardware-Erweiterungen

Heim- oder Kleincomputer stellen zumeist in sich relativ geschlossene Systeme dar. Wenn bei der Konzeption nicht schon von einem modularen Aufbau ausgegangen wurde, wie beim *KC 85* oder beim *Z 1013*, fällt eine hardwareseitige Erweiterung des Rechners dem mit der Rechentechnik weniger vertrauten Anwender oft schwer. Leider ist auch der *ZX Spectrum* bei der Ausstattung mit Schnittstellen mehr als stiefmütterlich behandelt worden. Außer dem Magnetbandanschluß, der ja im Grunde eine serielle Schnittstelle darstellt, und dem Anschluß für ein Fernsehgerät (leider auch nur über HF) sind in der Grundversion keine weiteren Anschlußmöglichkeiten vorhanden. Im folgenden soll daher gezeigt werden, wie es mit relativ einfachen Mitteln möglich ist, kleine Computer auch hardwareseitig zu erweitern. Dabei wurde Wert auf einfache, überschaubare Schaltungen gelegt, die zum größten Teil in Form von Bauanleitungen vorgestellt werden. Die Platinen liegen, soweit möglich, als einseitige Entwürfe vor, um den Nachbau zu erleichtern. Dafür müssen u. U. einige Drahtbrücken in Kauf genommen werden. Bei Notwendigkeit zweiseitiger Platinen wurden Lötinseln für Schaltkreise auf der Bestückungsseite konsequent vermieden. Damit ist im Falle eines Bauelementedefekts ein leichtes Entfernen des Schaltkreises ohne Beschädigung der Platine möglich. Die erforderlichen Durchkontaktierungen werden mit Drahtstücken, z. B. Anschlußdrähten von Widerständen, realisiert, die beidseitig zu verlöten sind.

Auf Verträglichkeit der Module mit dem weit verbreiteten ZX-Drucker wurde geachtet, obwohl mit der beschriebenen PIO-Karte und der entsprechenden Software eine Centronics- sowie eine V. 24-Schnittstelle zur Verfügung steht. Die Verträglichkeit mit dem Interface 1 und den aus heutiger Sicht als veraltet geltenden Microdrives ist nicht in jedem Fall gesichert. Allerdings fällt es dem erfahrenen Amateur sicher leicht, die Adreßselektion seinen Wünschen entsprechend zu ändern. Damit wäre dann auch eine Anpassung der Module an andere *U-880*-Computersysteme denkbar.

Soweit erforderlich und vom Umfang her möglich, wird die entsprechende Treibersoftware mit angegeben. Diese Programme stellen Minimalvarianten dar. Sie lassen damit breiten Spielraum für die Anpassung an die Aufgabenstellung beim Anwender.

5. 1. Der Erweiterungsanschluß des *ZX Spectrum*

Auf der Rückseite des *ZX Spectrum* stehen auf einem 2 x 28poligen direkten Steckverbinder alle für das Betreiben von Hardwarezusätzen erforderlichen Si-

Tabelle 5.1 Der Bussteckverbinder des *ZX Spectrum*

Signal	Anschluß		Signal
A15	1A	1B	A14
A13	2A	2B	A12
D7	3A	3B	+5 V
n. a.	4A	4B	+9 V
Verpol-schutz	5A	5B	Verpol schutz
D0	6A	6B	Masse
D1	7A	7B	Masse
D2	8A	8B	Takt
D6	9A	9B	A0
D5	10A	10B	A1
D4	11A	11B	A1
D3	12A	12B	A3
INT	13A	13B	IORQGE
NMI	14A	14B	Masse
HALT	15A	15B	VIDEO
MREQ	16A	16B	Y
IORQ	17A	17B	V
RD	18A	18B	U
WR	19A	19B	BUSRQ
-5 V	20A	20B	RESET
WAIT	21A	21B	A7
+12 V	22A	22B	A6
-12 V	23A	23B	A5
M1	24A	24B	A4
RFSH	25A	25B	ROMCS
A8	26A	26B	BUSAK
A10	27A	27B	A9
n. a.	28A	28B	A11

(n. a. - nicht angeschlossen)

gnale und Spannungen zur Verfügung. Der Steckverbinder besteht aus einer oberen (Bauteil-) und einer unteren (Löt-) Seite der Computerplatine. Die Anschlüsse tragen die Bezeichnung 1A bis 28A auf der Oberseite und 1B bis 28B auf der Unterseite (Tabelle 5.1).
Die Funktion der CPU-Signale wird als bekannt vorausgesetzt. Deshalb wird an dieser Stelle nur auf die Spannungsversorgung und die für den *ZX Spectrum* spezifischen Signale eingegangen:

-5 V (20A): Es liegt eine Spannung von -5 V an, die von einem internen Transverter erzeugt wird. Diese Spannung darf extern nur mit 5 mA belastet werden.

+12 V (22A): Diese Spannung wird ebenfalls von dem genannten

Transverter erzeugt. Sie darf mit höchstens 10 mA belastet werden.

-12 V (23A): Dieser Anschluß führt eigentlich keine -12 V, sondern eine hochfrequente Wechselspannung, aus der die -12 V aber mit einer Zusatzschaltung gemäß Bild 5.1 gewonnen werden können. Belastbar ist dieser Anschluß mit etwa 10 mA.

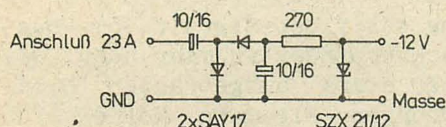


Bild 5.1 Schaltung zur Gewinnung der -12 V

+5 V (3B): An diesem Anschluß steht die intern stabilisierte Versorgungsspannung zur Verfügung. Diese Spannung kann extern mit etwa 200 mA belastet werden, bei höherer Stromentnahme kann es zu thermischen Problemen kommen.

+9 V (4B): Vom Netzteil gelieferte Rohspannung. Mit einer externen Stabilisierungsschaltung kann die Betriebsspannung für Verbraucher mit höherer Stromaufnahme (bis etwa 0,5 A) gewonnen werden.

IORQGE (13B): Dieses Signal stellt die logische AND-Verknüpfung der Prozessor-Signale $\overline{IORQ}$ und A0 dar. Es signalisiert einen aktivierten ULA-Schaltkreis.

VIDEO, Y, V, U (15B bis 18B): Diese Signale stehen im Zusammenhang mit der Bildsignalaufbereitung des *ZX Spectrum*.

ROMCS (ROM CHIP SELECT, 25B): Bei H-Pegel an diesem Anschluß wird das interne ROM abgeschaltet. Es können dann externe ROMs oder RAMs mit anderen Programmen eingeblendet werden.

5.2. Reset-Taste

Die Reset-Taste legt bei Betätigung den low-aktiven RESET-Eingang des *Z 80* auf Masse. Mit der L/H-Flanke, also bei Loslassen der Taste, beginnt der Prozessor (neben einigen anderen Aktivitäten) mit dem Abarbeiten des Programms ab Adresse 0000.

Dadurch wird das gesamte im RAM befindliche Programm gelöscht, da im Betriebssystem des *Spectrum* eine Initialisierungsroutine eingebaut ist. Insofern ist diese Taste nur als äußerste "Notbremse" zu verstehen, wenn der Rechner in einer endlosen Schleife festhängt und die Tastaturabfrage nicht mehr funktioniert. Immerhin dient sie aber der Schonung des schwer beschaffbaren Hohlsteckers für die 9-V-Versorgung, mit dem bisher Reset ausgelöst wurde.

Bei Ersatz des internen ROM durch einen EPROM kann die genannte Routine vorteilhaft entfernt werden. Seit einiger Zeit sind auch sogenannte ISO-ROMs verfügbar, die mit Reset ohne Programmverlust arbeiten.

Zwischen dem Reset-Anschluß der CPU und Masse ist ein Elektrolytkondensator angeordnet, der in Verbindung mit einem Widerstand für das Einschaltreset sorgt. Dieser Kondensator wird beim Betätigen der Reset-Taste

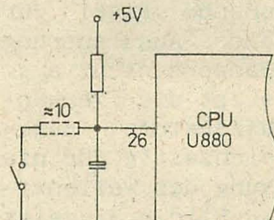


Bild 5.2 Anschluß einer RESET-Taste

kurzgeschlossen. Es ist deshalb empfehlenswert, in Reihe mit der Reset-Taste einen Widerstand von 10 bis 100 Ω zur Strombegrenzung zu schalten. Bei Einsatz einer Elastomertaste (z. B. TSE 15) ist das nicht erforderlich, da deren Kontaktwiderstand im geschlossenen Zustand nicht Null ist.

5. 3. Video-Anschluß

Das direkte Einspeisen des Video-Signals in einen Fernsehempfänger ergibt ein wesentlich schärferes und kontrastreicheres Bild, da der gesamte HF-Trakt sowohl im Fernsehgerät als auch im Computer umgangen wird. Dazu ist ein kleiner Eingriff in das Fernsehgerät notwendig, der von einem Fachmann oder versierten Amateur auszuführen ist. Volltransistorisierte Fernsehempfänger bieten i. allg. die Gewähr der Netztrennung. Allstromgeräte sind aus Sicherheitsgründen unbedingt über einen Trenntransformator zu betreiben! Sollte das Video-Signal des *ZX Spectrum* auf dem Bussteckverbinder nicht nachweisbar sein, ist im Inneren des Geräts eine Drahtbrücke nachzurüsten, die im Bestückungsdruck der Platine gekennzeichnet ist. Die einfachste Variante ist die direkte Verwendung des Video-Signals vom Anschluß 15B. Oft reicht jedoch der dort anstehende Pegel für eine ausreichende Aussteuerung des Fernsehempfängers nicht aus. In diesem Falle ist die Verwendung einer Pufferstufe angebracht, die gleichzeitig für die richtige Anpassung des Wellenwiderstands sorgt. Bild 5. 3 zeigt eine solche Schaltung. In manchen Fällen kommt es bei Verwendung des Video-Signals vom Bussteckverbinder zu mehr oder weniger stark ausgeprägten Bildstörungen (Flimmern, Moiré). Hier schafft oft die Auskopplung des Video-Signals direkt am Modulator schon Abhilfe, manchmal kann jedoch Abschalten des Farbträgers nötig werden, wie es in [7] beschrieben wurde.

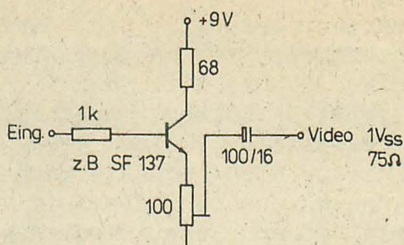


Bild 5. 3 Pufferstufe für das Video-Signal

5. 4. Lichtstift

Als Lichtstifte oder Lichtgriffel werden Computer-Zusatzgeräte bezeichnet, die in Verbindung mit einem entsprechenden Steuerprogramm das Auslösen von Reaktionen durch Zeigen auf bestimmte Stellen des Computermonitors ermöglichen. Wenn es Elektronik und Steuerprogramm erlauben, die Positionsbestimmung des Lichtstiftes während eines Bildaufbaus zu bewerkstelligen, kann mit dem Lichtstift auf dem Moni-

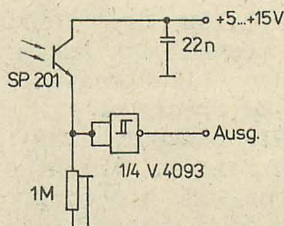


Bild 5. 4 Schaltung des Lichtstiftes mit CMOS-Gatter



Bild 5.5 Platine für die Schaltung nach Bild 5.4

tor z.B. wahlfrei gezeichnet werden. Die hier beschriebenen beiden Schaltungen halten in Verbindung mit dem verwendeten Programm diesen Echtzeitforderungen nicht stand. Die Positionsbestimmung erfordert etwas mehr Zeit, dennoch läßt sich die Wirkungsweise von Lichtstiften damit gut demonstrieren. Die Schaltung eines Lichtstiftes für die beschriebenen geringen Anforderungen ist sehr einfach aufgebaut. Sie besteht aus einem einzigen Gatter eines CMOS-Schmitt-Trigger-NAND, an den ein Fototransistor mit einem Widerstand zur Arbeitspunkteinstellung angeschlossen ist.
Stromlaufplan, Leiterbild und Bestückungsplan sind in den Bildern 5.4 bis 5.6 dargestellt. Die Leiterplatte ist so gestaltet, daß sie in ein Gehäuse eines leeren Faserschreibers (Steppke-Stift oder Pfau-Dubbel) eingebaut werden kann. Angeschlossen wird der Lichtstift an die EAR-Buchse des *Spectrum*, wodurch der Busstecker für weitere Anwendungen frei bleibt.
Die Spannungsversorgung kann vorteilhaft durch eine Flachbatterie erfolgen, natürlich sind die stabilisierten 5 V des *Spectrum* ebenso geeignet. Die Funktion der Schaltung ist wegen der geringen Anzahl der Bauelemente leicht zu verstehen. Der Eingang des CMOS-Gatters liegt durch das Trimmpotentiometer bei unbeleuchtetem Fototransistor auf L-Potential, sein Ausgang damit auf H. Bei beleuchtetem Fototransistor wird dieser leitend und die Verhältnisse kehren sich um. Zur Begrenzung des Ausgangsstroms des Gatters sind im Zusammenspiel mit dem *Spectrum* keine weiteren Maßnahmen erforderlich. Sollte für andere Anwendungen eine Strombegrenzung nötig werden, ist der mit einem Pfeil gekennzeichnete Leiterzug auf der Platine zu durchtrennen und ein entsprechender Widerstand einzufügen. Die unbenutzten Gattereingänge sind auf definiertes Potential zu legen.
Eine ebenfalls mögliche, aber nicht ganz so empfindliche Lichtstift-Schaltung mit dem Initiatorschaltkreis *A 301 D* zeigt das Bild 5.8. Der Widerstand zwischen den Anschlüssen 2 und 4 zur Festlegung der Hysterese kann gegebenenfalls auch entfallen. Das ist in Abhängigkeit vom verwendeten Monitor und der eingestellten Grundhelligkeit experimentell zu ermitteln.
In beiden Schaltungen kann der



Bild 5.6 Bestückungsplan der Platine nach Bild 5.5



Bild 5.7 Musteraufbau des Lichtstiftes

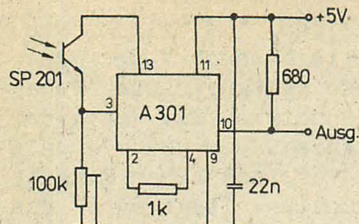


Bild 5.8 Lichtstiftschaltung mit A 301

Fototransistor durch lichtundurchlässigen Lack etwas gegen seitlich einfallendes Fremdlicht geschützt werden. Das Trimmpotentiometer ist bei gestartetem Programm "Lpen 48" in die Mitte des Arbeitsbereichs zu stellen. Das ist der Bereich, in dem die Pfeile auf dem Bildschirm blinken. Die weitere Kalibrierung erfolgt nach den vom Programm gegebenen Hinweisen. Vor Anschluß der Betriebsspannung ist das Potentiometer auf größten Widerstandswert zu stellen! Das Programm "Lpen 48" würde aufgrund seines Umfangs den Rahmen dieses Buches sprengen. Es ist aber mittlerweile so weit verbreitet, daß ein Abdruck nicht unbedingt nötig erscheint. Deshalb wird an dieser Stelle nur die Programmbeschreibung gegeben. Ein Bildschirmpunkt wird angewählt, indem der Lichtstift an diese Stelle gehalten und eine beliebige Taste gedrückt wird. Daraufhin wird das große Kreuz an diese Stelle bewegt.
Die Optionen des Bildschirmmenüs werden ebenso angewählt. In dieser Beschreibung wird das große Kreuz mit + und das kleine Kreuz mit x bezeichnet.

Bedeutung der Optionen:

A (arc) zeichnet einen Bogen von x durch die letzte Position von + zur aktuellen Position von +
B (border) BORDER-Farbe ändern
C (circle) zeichnet einen Kreis durch + mit Mittelpunkt x
D (draw) zeichnet eine Linie zwischen beiden Kreuzen
E (erase) löscht die letzte Aktion
F (fill) füllt ein umrandetes Gebiet, in dem sich + befindet, mit INK-Farbe
H (hand) erlaubt beliebiges Zeichnen mit dem LP
I (ink) INK-Farbe ändern
K (keep) bedeutet, daß bis zu vier Bildschirme im RAM abgespeichert werden können
L (letters) schreibt Text ab der Stelle +
M (move) bewegt das kleine Kreuz x auf das große Kreuz +
N (new) löscht den Bildschirm
P (paper) PAPER-Farbe ändern
R (rectangle) zeichnet ein Rechteck zwischen den Kreuzen
R (restore) bedeutet, gespeicherte Bilder können einzeln oder zyklisch abgerufen werden
T (tape) bedeutet, Bildschirme können vom/auf Band geladen werden.

5.5. Joystick-Anschluß

Spielhebel oder Joysticks sind zumindest für Kleincomputer fast unentbehrliche Zusatzgeräte. Sie lassen sich sehr vielseitig verwenden, obwohl ihr Haupteinsatzgebiet die Steuerung von Computerspielen sein dürfte.
Während die Realisierung eines Joysticks an sich ein rein mechanisches Problem ist, erfordert der Anschluß des Joysticks an den Computer zumeist etwas Zusatzhardware, die in der Grundversion nicht in allen Kleincomputern enthalten ist. Im folgenden werden einige Varian-

ten des Joystick-Anschlusses an den *ZX Spectrum* vorgestellt, die sinngemäß auch für andere Computer-Typen verwendet werden können.

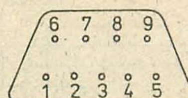


Bild 5.9 Genormte Anschlußbelegung des Joystick-Steckers

5.5.1. Joystick-Anschluß ohne Zusatzhardware

In den meisten Fällen ist die Steuerung der Bewegungsabläufe auf dem Bildschirm auch über die Tastatur möglich. Es liegt daher nahe, den Joystick einfach den entsprechenden Tasten parallel zu schalten. Diese Methode erfordert zwar genaue Kenntnisse über die Anordnung der Tasten in der Tastaturmatrix und darüber hinaus einen Eingriff in den Rechner, bietet aber dafür den Vorteil kaum zu übertreffender Einfachheit.

Bild 5.9 zeigt die inzwischen international genormte Belegung des Steuerknüppel-Anschlusses. Daraus geht hervor, daß neben 4 Anschlüssen für die Richtungswahl und einem für den Trigger ("Schußknopf") nur eine gemeinsame Leitung vorhanden ist. Bei der Parallelschaltung zur Tastatur ist dem Rechnung zu tragen, so daß die verwendeten Tasten alle an einer gemeinsamen Spalten- bzw. Zeilenleitung angeschlossen sein müssen. Im Bild 5.10 ist die Tastaturmatrix des *ZX Spectrum* wiedergegeben. Aus dem eben erwähnten Grund ist eine Parallelschaltung zu den Cursor-Tasten 5 bis 8 nicht möglich. Deshalb werden die Tasten 6 bis 9 für die Richtungssteuerung und die Taste 0 für den Trigger verwendet. Auf diese Weise ist sogar ein zweiter Joystick anschließbar, der den Tasten 1 bis 5 zugeordnet wird. Bild 5.11 stellt die Verbindung der beiden Joystick-Buchsen mit den Tastatursteckverbindern auf der Platine dar.

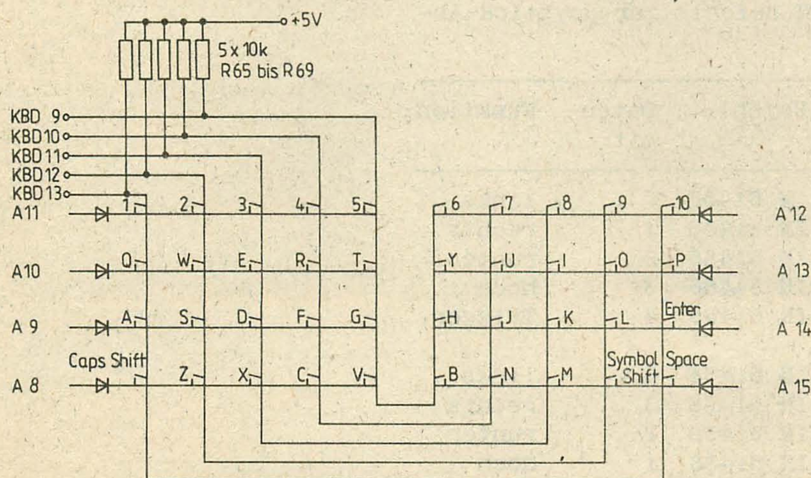


Bild 5.10 Tastaturmatrix des ZX Spectrum

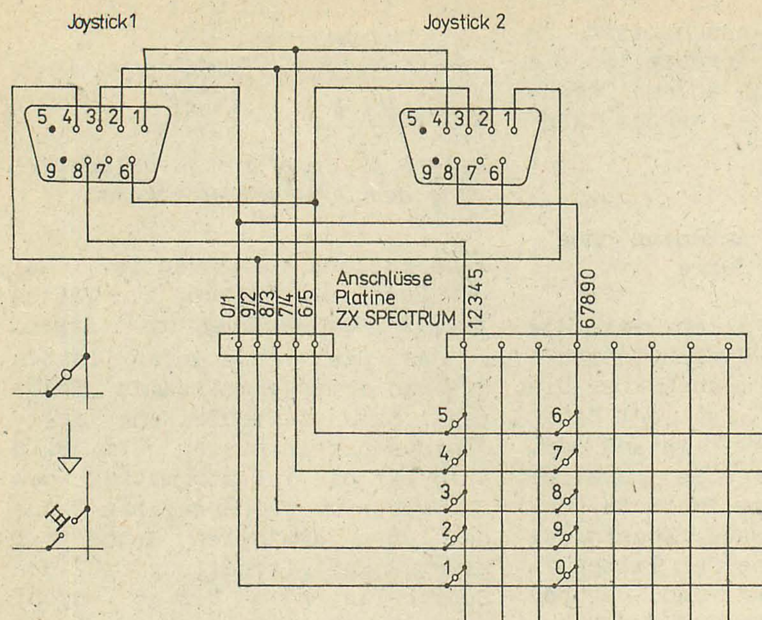


Bild 5.11 Direktanschluß zweier Joysticks an die ZX Spectrum-Tastatur

Die Abfrage der Steuerknüppel geschieht mit zwei IN-Befehlen in Basic oder Maschinencode; der Stellung der Joysticks entsprechend werden dabei die Datenbits 0 bis 4 gesetzt und gleichzeitig wahrgenommen. Damit ist also die Realisierung relativ schneller Grafik-Funktionen möglich. Tabelle 5.2 zeigt das noch einmal ausführlich. Bei kommerzieller Software wird oft ein Menü mit

Tabelle 5.2 IN-Befehle zur Joystick-Abfrage

Joystick	Taste	Befehl	Datenbit	Funktion
1	1	IN 61486	0	links
1	2	IN 61486	1	rechts
1	3	IN 61486	2	runter
1	4	IN 61486	3	hoch
1	5	IN 61486	4	Trigger
2	6	IN 61438	4	links
2	7	IN 61438	3	rechts
2	8	IN 61438	2	runter
2	9	IN 61438	1	hoch
2	0	IN 61438	0	Trigger

verschiedenen Joystick-Normen angeboten. Für die vorgestellte Anschlußvariante ist die Option *Sinclair* zu wählen.

5.5.2. Joystick-Anschluß über Peripherie-Schaltkreise

Trotz des geringen Aufwands bei der Parallelschaltung des Joysticks zur Rechnertastatur gibt es einige Gründe, die für einen Anschluß der Spielhebel über eine Anpaßschaltung sprechen. Das sind unter anderem:
- gemeinsamer Anschluß des Joysticks liegt auf definiertem Potential,
- verschiedene Anschlußnormen sind nur so realisierbar,
- Eingriff in den Rechner entfällt.

Im folgenden wird deshalb gezeigt, wie ein Joystick über ein Register an den Datenbus des *ZX Spectrum* angeschlossen werden kann. Dieses Register wird bei einer bestimmten Kombination von Adressen und Steuersignalen des Prozessors aktiviert. Dazu dient ein Adreßdekoder.
Bild 5.12 zeigt das Prinzip eines Joystick-Anschlusses entsprechend der *Sinclair*-Norm. Dieses Interface bildet im Prinzip die Tastaturabfrage für die interessierenden Tasten

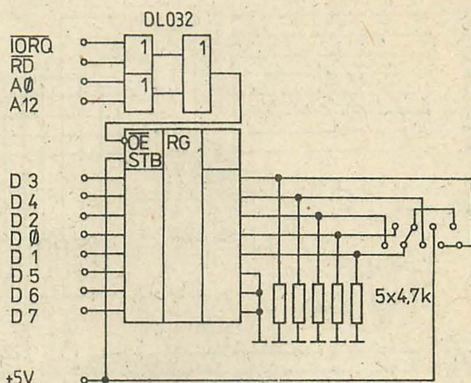


Bild 5.12 Joystick-Interface (Sinclair-Norm)

nach, indem es die Steuersignale $\overline{IORQ}$ (Ein-/Ausgabe-Anforderung), $\overline{RD}$ (Prozessor will Daten lesen) und die Adressen A0 (ULA aktiv) und A12 (Tastatur-Reihe 6 bis 0) auswertet und bei gleichzeitiger Aktivierung das Register selektiert. Damit wird ein Bitmuster, das der Stellung des Joysticks entspricht, auf den Datenbus gelegt.

5.5.3. *Kempston*-Joystick

Für den Anschluß von Joysticks an den *ZX Spectrum* gibt es verschiedene Normen, von denen wohl die *Kempston*-Norm die weiteste Verbreitung gefunden hat. Diese Norm legt fest, daß die Stellung des Joysticks über die Port-

Tabelle 5.3 IN-Befehle zur Abfrage des *Kempston*-Joysticks

Befehl	Datenbit	Funktion
IN 223	0	links
IN 223	1	rechts
IN 223	2	runter
IN 223	3	hoch
IN 223	4	Trigger

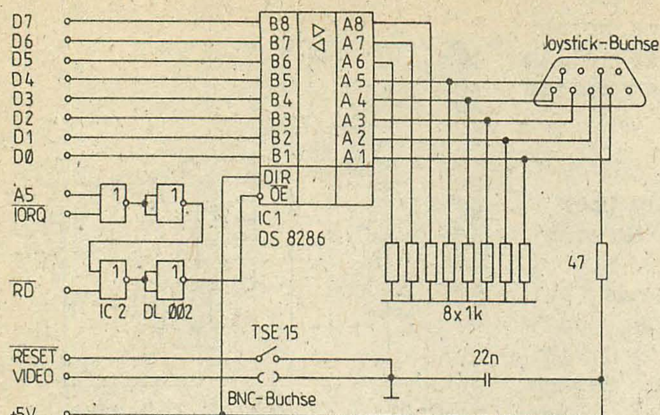


Bild 5.13 Schaltung eines Kempston-Joystick-Interfaces

adresse DFH (223 dezimal) einlesbar ist und dort durch gesetzte Bits 0 bis 4 repräsentiert wird. Die Zuordnung der gesetzten Bits zur Stellung des Joysticks kann der Tabelle 5.3 entnommen werden.

Das *Kempston*-Interface besteht ebenfalls aus einem Register und einem Adreßdekoder. Als Register kommt hier der Bustreiber-Schaltkreis *DS 8286* zum Einsatz. Dessen Mikroprozessorseite (Anschlüsse 1 bis 8) und Datenbusseite (Anschlüsse 12 bis 19) wurden für diese Anwendung vertauscht. Dadurch ist es möglich, an dieser Stelle auch den Typ *DS 8282* einzusetzen. Da die Datenübertragung beim Joystick-Anschluß generell nur in der Richtung Peripherie ---> Prozessor erfolgt, spielt die geringere Stromergiebigkeit des *DS 8286* auf der Mikroprozessorseite keine Rolle. Allerdings ist die typische Stromaufnahme des *DS 8286* etwa um 20 mA geringer als beim Typ *8282*. Bei Änderung der Platine können natürlich auch andere Register mit Tri-state-Ausgängen verwendet werden.

Erfolgreich getestet wurde in diesem Zusammenhang auch der CMOS-Treiber *V 40098*. Zu beachten ist dabei, daß dieser

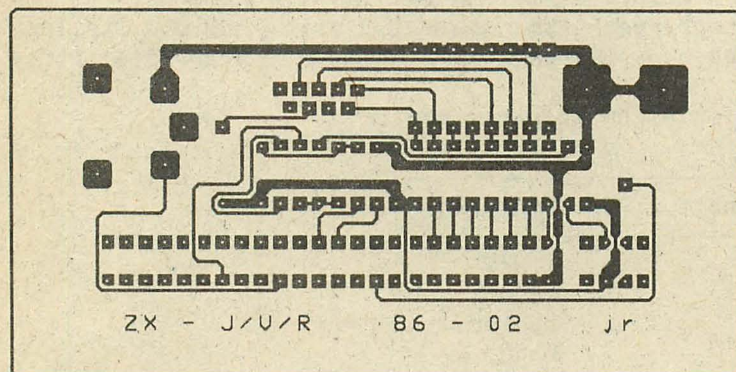


Bild 5.14 Layout für das Kempston-Interface

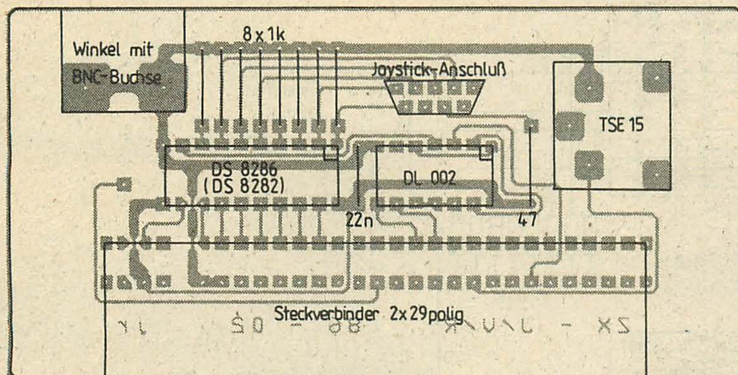


Bild 5.15 Bestückungsplan der Platine nach Bild 5.14

Schaltkreis die Daten invertiert. Deshalb ist der gemeinsame Anschluß des Joysticks an Masse zu legen. Das gilt auch bei Verwendung der invertierenden Bustreiber *DS 8283* und *DS 8287*.

Als Adreßdekoder arbeitet in dieser Schaltung ein *DL 002 D*. Er wurde aufgrund seiner besseren Verfügbarkeit dem *DL 032 D* vorgezogen. Der Dekoder selektiert das Register immer dann, wenn der Prozessor eine Eingabeanforderung bezüglich der Adresse 5 sendet (IORQ, RD und A5 führen Low-Pegel). Somit wird, bedingt durch die Widerstände R2 bis R9, das Datenwort 00H an den Prozessor gegeben, sofern der Joystick unbetätigt ist. Im Falle einer Betätigung führen die entsprechenden Leitungen über R1 H-Pegel. Für diese Anwendung würde auch ein 5-bit-Register ausreichen. Das Einlesen von 8 bit (bit 5 bis 7 sind dann immer 0) bietet aber die Gewähr, daß auch Programme, die nicht nur die unteren 5 bit auswerten, mit dem Interface zusammenarbeiten können. Neben dem Joystick-Interface wurden auf der Platine noch eine Reset-Taste und ein Video-Anschluß untergebracht. In den Bildern 5.13 bis 5.15 sind Schaltung, Platinen-Layout und Bestückungs-

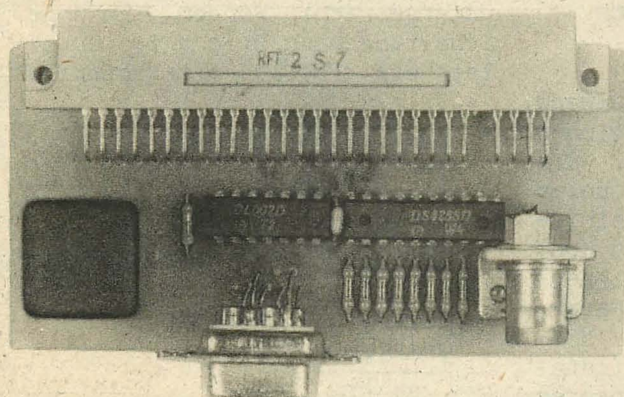


Bild 5.16 Musteraufbau des Kempston-Interfaces

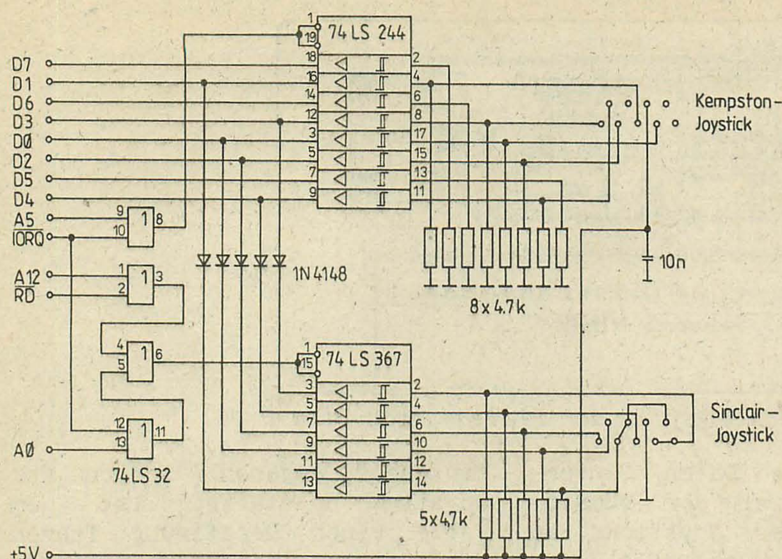


Bild 5.17 dk'tronics-Interface

plan des *Kempston*-Interfaces dargestellt; im Bild 5.16 ist das aufgebaute Muster abgebildet. Bild 5.17 zeigt die Schaltung eines Joystick-Interfaces der Firma *dk'tronics*. Es realisiert unter Verwendung von zwei Registern einen *Kempston*- und einen *Sinclair*-Anschluß. Für den Schaltkreis *74 LS 367* kann der pinkompatible *V 40098* verwendet werden.

5.5.4. Zusatzfunktionen für Joysticks

Mit geringem Hardwareaufwand sind 2 Erweiterungsfunktionen für Joysticks realisierbar, die insbesondere den Spielern unter den Computerbesitzern einige Erleichterungen verschaffen können. Es sind dies ein Triggergenerator (landläufig als Dauerfeuer bezeichnet) und ein NMI-Generator (aus der internationalen Literatur auch als Freezer bekannt).

Der Triggergenerator wird zwischen Joystick und Anpaßelektronik geschaltet. Er erzeugt bei Drücken der Triggertaste am Joystick eine Impulsfolge, deren Frequenz bei beiden vorgestellten Schaltungen zwischen 2 und etwa 25 Hz einstellbar ist.
Wegen des geringen Umfangs der Schaltungen lohnt die Anfertigung einer speziellen Platine kaum. Die Elektronik kann, aufgebaut auf einer Universalplatine, direkt im Joystickgehäuse untergebracht werden. In der Schaltung nach Bild 5.18 arbeitet ein *V 4093* als astabiler

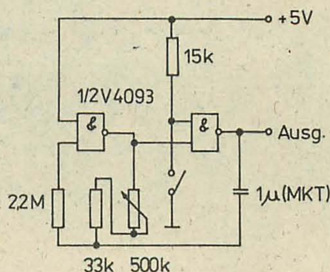


Bild 5.18 CMOS-Triggergenerator

Multivibrator. Für den 1-µF-Kondensator sollte kein Elektrolytkondensator verwendet werden. Die Stromaufnahme dieser Schaltung beträgt etwa 1 mA. Das ist ein Wert, der (für andere Anwendungen) auch viele Stunden Betrieb mit einer Flachbatterie garantiert.

In Bild 5.19 ist ein *B 555* ebenfalls als astabiler Multivibrator beschaltet. Die Diode zwischen den Anschlüssen 6 und 7 ermöglicht den Angleich der H- und L-Zeiten. Die beiden Schaltungen benötigen keinen Abgleich, können aber durch Änderung der zeitbestimmenden Bauelemente den Erfordernissen des Anwenders angepaßt werden. Je nach Potential des gemeinsamen Anschlusses am Joystick wird der Generator über die positive Betriebsspannung oder die Masseleitung eingeschaltet. Bei Verwendung des Joystick-Anschlusses nach Abschnitt 5.6.1. können diese Schaltungen nicht verwendet werden, da hier der gemeinsame Joystick-Anschluß nicht auf Masse oder $+U_b$ liegt.

Im allgemeinen strebt man danach, daß der Computer seine Aufgaben so schnell wie möglich ausführt. Zu diesem Zweck wird, insbesondere bei größeren Systemen, der Mikroprozessor durch entsprechende Zusatzhardware

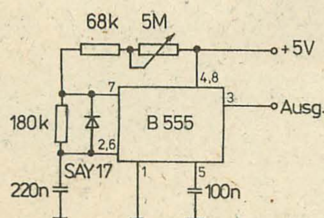


Bild 5.19 Triggergenerator mit B 555

(z. B. Arithmetik-Prozessor, Video-Controller) entlastet.

Auch eine Erhöhung der Taktfrequenz zielt in diese Richtung, erfordert aber Bauelemente, die den kürzeren Taktzeiten gerecht werden. In einigen Fällen kann es aber durchaus sinnvoll sein, die Aktivitäten eines Computers auch zu verlangsamen, um etwa den Aufbau einer komplizierten Grafik pixelweise zu verfolgen, Bewegungsabläufe auf dem Bildschirm zu studieren oder in Computerspielen Vorteile durch längere Reaktionszeit zu erreichen. Für diese Zwecke ist zunächst eine Verringerung der Taktfrequenz denkbar. Allerdings bringt das mit Sicherheit einige Funktionen des Computers durcheinander, da z. B. Zeitschleifen zur Synchronisation interner Abläufe für die Normal-Taktfrequenz berechnet sind. Deshalb wird hier eine andere Möglichkeit angeboten.

Um den Prozessor *U 880* zu zwingen, seine derzeitige Arbeit zugunsten eines anderen Programmes zu unterbrechen, hat er 2 Eingänge, die mit $\overline{INT}$ und $\overline{NMI}$ bezeichnet sind. Der Eingang $\overline{INT}$ wird mit Low-Pegel aktiviert, der Eingang $\overline{NMI}$ durch eine HL-Flanke. Während eine am Eingang $\overline{INT}$ anliegende Interruptanforderung durch die Software erlaubt oder abgewiesen (maskiert) werden kann, ist das beim Eingang $\overline{NMI}$ (deshalb Nicht Maskierbarer Interrupt) nicht möglich. Ein $\overline{NMI}$ zwingt den Prozessor *U 880* zu einem RESTART ab Speicherplatz 66H. Der Befehlszähler wird automatisch in den externen Stack gerettet, so daß der Anwender nach Abarbeitung der NMI-Service-Routine zum unterbrochenen Programm zurück-

kehren kann. Dazu dient der Maschinenbefehl RETN (RETurn vom NMI). Tabelle 5.4 zeigt die NMI-Service-Routine des *ZX Spectrum*. Offensichtlich war diese Routine ursprünglich dazu gedacht, das Programm in Abhängigkeit vom Inhalt der Systemvariablen NMIREG zu verzweigen. Durch einen kleinen Programmfehler ist das aber nicht möglich. Abhängig vom Inhalt der Systemvariablen NMIREG kann man lediglich einen Kaltstart entsprechend RESET ausführen (bei NMIREG = 0) oder nichts passieren lassen (bei NMIREG nicht = 0). Die Abarbeitung des NMI-Programms erfordert für letztgenannten Fall 92 Prozessortakte. Bei 3,5 MHz Taktfrequenz beträgt die Länge einer Taktperiode etwa 286 ns. Daraus läßt sich die Laufzeit der NMI-Routine errechnen:
92•286 ns = 26,3 µs. Wird der Prozessor gezwungen, unmittelbar nach Ablauf der NMI-Routine einen neuen NMI auszuführen, bleibt ihm keine Zeit mehr für das Abarbeiten seines eigentlichen Programms. Es kommt somit zum Stillstand. Für praktische Anwendungen ist es sinnvoll, diesen "NMI-Generator" stetig oder in Stufen einstellbar auszuführen. Damit ist es möglich, nicht so interessante Programmteile in normaler Geschwindigkeit abzuarbeiten.
Um einen NMI alle 26,3 µs auszulösen, ist theoretisch eine Impulsfolgefrequenz von 38 kHz erforderlich. Praktisch genügt schon eine etwas geringere Frequenz, da das ULA im *ZX Spectrum* hin und wieder den Takt anhält, was einer geringeren mittleren Taktfrequenz gleichzusetzen ist. Im Mustergerät wurde als obere Grenzfrequenz 32,5 kHz ermittelt. Eine Erhöhung der NMI-Impulsfolge über diese Frequenz hinaus führt zum Systemabsturz, da durch die am Anfang der NMI-Routine abgearbeiteten PUSH-Befehle der Stack den gesamten RAM-Bereich überschreibt. Vor Inbetriebnahme des NMI-Generators muß zumindest in eine der Systemvariablen 23728 und 23729 ein Wert eingetragen werden, der von 0 verschieden ist. Diese Systemvariablen sind im Handbuch fälschlich als nicht benutzt angegeben.
Im Interesse einer möglichst geringen Leistungsaufnahme des NMI-Generators werden eine CMOS-

Tabelle 5.4 NMI-Routine des *ZX Spectrum*

Adresse	Code	Marke	Befehl
0066	F5	NMI	PUSH AF
0067	E5		PUSH HL
0068	2A B0 5C		LD HL, (NMIREG)
006B	7C		LD A, H
006C	B5		OR L
006D	20 01		JR NZ, #70
006F	E9		JP (HL)
0070	E1		POP HL
0071	F1		POP AF
0072	ED 45		RETN

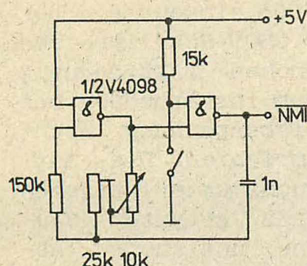


Bild 5.20 CMOS-NMI-Generator

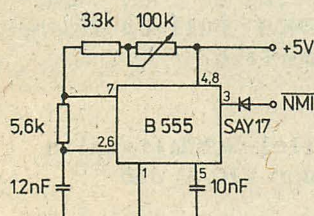


Bild 5.21 NMI-Generator mit B 555

und eine *B-555*-Version vorgestellt.
Bild 5.20 zeigt die CMOS-Version des NMI-Generators. In dieser Schaltung arbeitet ein *V 4011* als astabiler Multivibrator mit veränderlicher Frequenz. Mit dem Widerstand R1 ist bei auf größten Widerstandswert eingestelltem Potentiometer die Frequenz auf 32,5 kHz einzustellen. Der Generator kann mit dem an einem Gattereingang befindlichen Schalter außer Betrieb gesetzt werden. Der Ruhepegel (H oder L) hängt davon ab, welches Gatter dafür verwendet wird. Da der NMI flankengetriggert ist, sind beide Ruhepegel möglich. Im Interesse besserer Flankensteilheit kann für den *V 4011* auch der pinkompatible *V 4093* eingesetzt werden. Bei Verwendung eines (allerdings nicht pinkompatiblen !) NOR-Gatters *V 4001* müssen die unbenutzten Gattereingänge an Masse gelegt werden. Die beiden anderen Gatter können vorteilhaft zur Realisierung der Dauerfeuerfunktion verwendet werden, so daß für beide Zusatzschaltungen nur ein Schaltkreis erforderlich ist. Im Bild 5.21 ist ein NMI-Generator mit dem *B 555* dargestellt. Diese Schaltung erzeugt mit der vorgeschlagenen Dimensionierung Frequenzen zwischen 3 und 30 kHz. Die Frequenzkonstanz ist etwas besser als bei der CMOS-Variante, allerdings ist auch die Außenbeschaltung umfangreicher. Die Stromaufnahme liegt bei 5 mA. Durch die Verwendung des Doppeltimers *B 556* sind auch hier beide Zusatzschaltungen mit einem Schaltkreis realisierbar.
Die Verwendung von Generatoren mit TTL oder anderen Logikfamilien ist natürlich ebenso möglich, bietet aber gegenüber den beschriebenen keine entscheidenden Vorteile. Denkbar sind auch "intelligente" Lösungen, z. B. die Verwendung eines CTC-Schaltkreises in der Betriebsart Zeitgeber.

5.6. Parallel-Schnittstellen

5.6.1. Parallel-Schnittstellen mit Registerschaltkreisen

Unter Nutzung von Bustreiberschaltkreisen, Auffangregistern (Latches) oder gemeinsam getakteten Flip-Flops ist der Aufbau eines parallelen Ein-/Ausgabeports sehr einfach möglich. Mit Bustreiberschaltkreisen entsteht dabei sogar der Vorteil einer weitaus höheren Stromergiebigkeit des Ausgabeports gegenüber der Verwendung von Standard-Peripheriebausteinen. Der Einsatz solcher Ports bietet sich

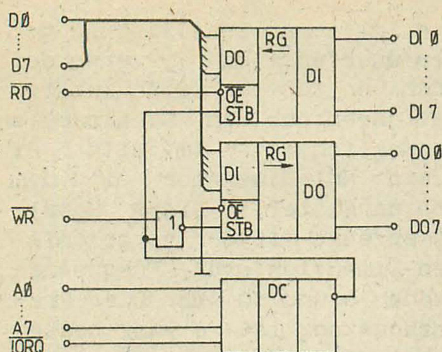


Bild 5.22 Parallelport mit zwei Bustreiberschaltkreisen (Prinzip)

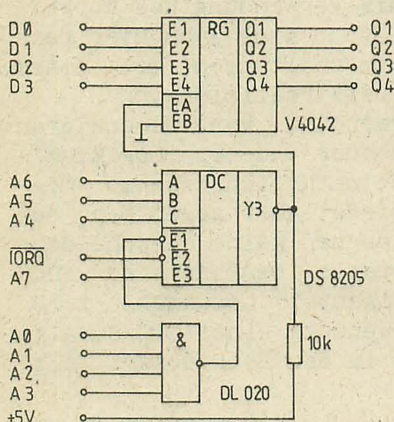


Bild 5.23 4-bit-Parallelport mit V 4042

also immer dann an, wenn die Programmierbarkeit der Eigenschaften der Schnittstelle nicht zwingend notwendig ist. Die Schaltung nach Bild 5.22 stellt unter Einsatz von 2 Bustreiberschalkreisen jeweils 8 Ein- und Ausgänge bereit. Dabei wird nur eine Adreßleitung in die Adressierung der Schnittstelle einbezogen (Stichleitungsprinzip). Im Bild 5.23 ist eine 4 bit breite Parallelschnittstelle unter Verwendung eines CMOS-Registers dargestellt. Die Adreßselektion ist durch Verwendung des Adreßdekoders *DS 8205* eindeutig. Der Einsatz von CMOS-Schaltkreisen ist bei der hohen Taktfrequenz des *ZX Spectrum* insbesondere bei nur 5 V Betriebsspannung nicht generell zu empfehlen. Die aufgebauten Versuchsmuster zeigten aber bei Einsatz getypter Bauelemente keine unsichere Arbeitsweise. Zum sicheren Erreichen des H-Pegels der CMOS-Schaltkreise sind an den Übergängen TTL-CMOS Pull-up-Widerstände erforderlich.

5.6.2. Parallel-Schnittstelle mit der PIO *U 855*

Die Schaltung im Bild 5.24 zeigt den Anschluß einer PIO *U 855* an den *ZX Spectrum*. Wegen der Vielzahl der Verbindungen und der relativ willkürlichen Anordnung der PIO-Anschlüsse ist eine zweiseitige Leiterplatte erforderlich. Die Leiterzüge auf der Bestückungsseite sind aber alle horizontal und parallel angeordnet, so daß sie sich auch durch Drahtbrücken ersetzen lassen.
Wie auch bei den anderen vorgestellten Baugruppen wird eine vollständige Adreßdekodierung verwendet. Unter Fortfall des Adreßdekoders IS2 kann jedoch auch das Stichleitungsprinzip zur Anwendung kommen. Dazu ist die auf der Bestückungsseite der Platine gekennzeichnete Drahtbrücke einzusetzen. Das gilt besonders dann, wenn keine weiteren Zusatzbaugruppen am Rechner betrieben werden. Der Adreßdekoder wird über den low-aktiven Eingang $\overline{E1}$ nur dann freigegeben, wenn das ULA (Adresse A0) und der ZX-Printer (Adresse A1) nicht angesprochen werden. Erreicht wird das durch ein Gatter

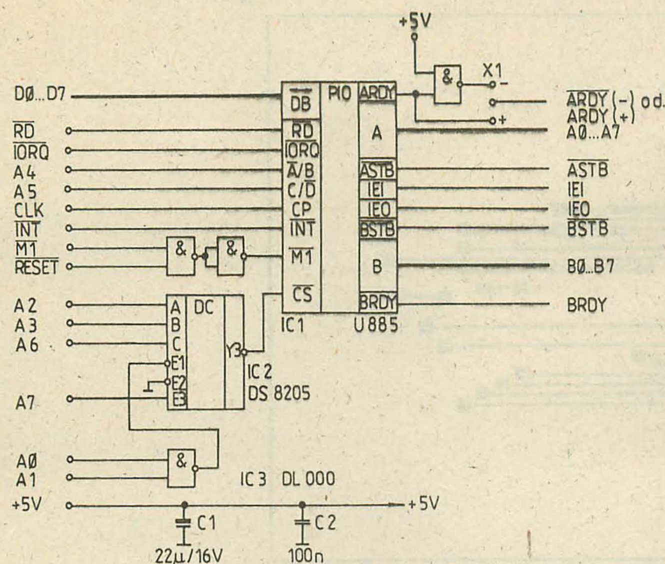


Bild 5. 24 PIO-Anschluß an den ZX Spectrum

des Schaltkreises IS3. Die Adreßleitungen A2, A3, A6 und A7 sind an den Dekoder geführt und bilden den Geräteauswahlimpuls (CS) für die PIO. Die Adressen A4 und A5 sind direkt mit der PIO verbunden. Sie dienen zur Auswahl von Kanal A oder B bzw. zur Interpretation der empfangenen Daten als Steuer- oder Datenwort. Nach Einschalten der Betriebsspannung geht die PIO *U 855* automatisch in den RESET-Zustand (Power-on-Reset). Dieser Zustand ist gekennzeichnet durch:

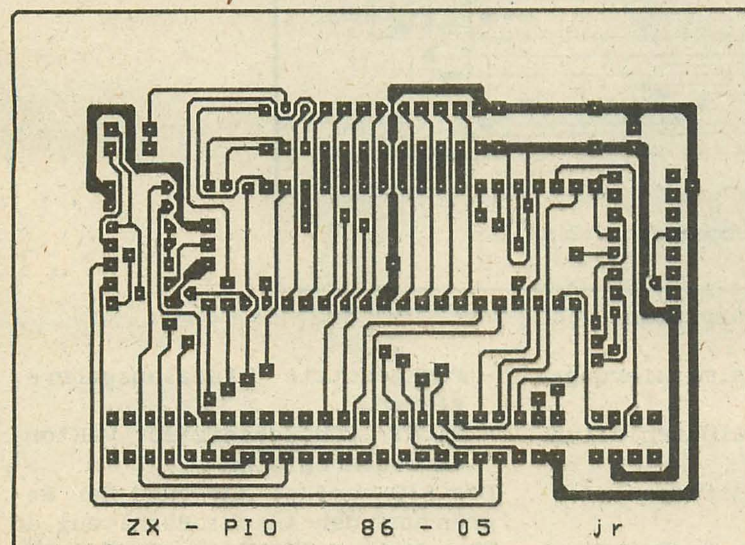


Bild 5. 25 Layout (Leiterseite) für die PIO-Karte

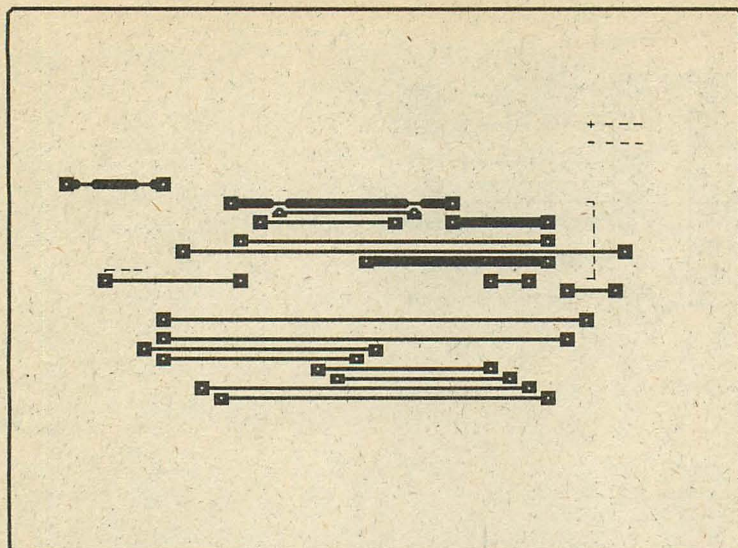


Bild 5.26 Layout (Bestückungsseite)

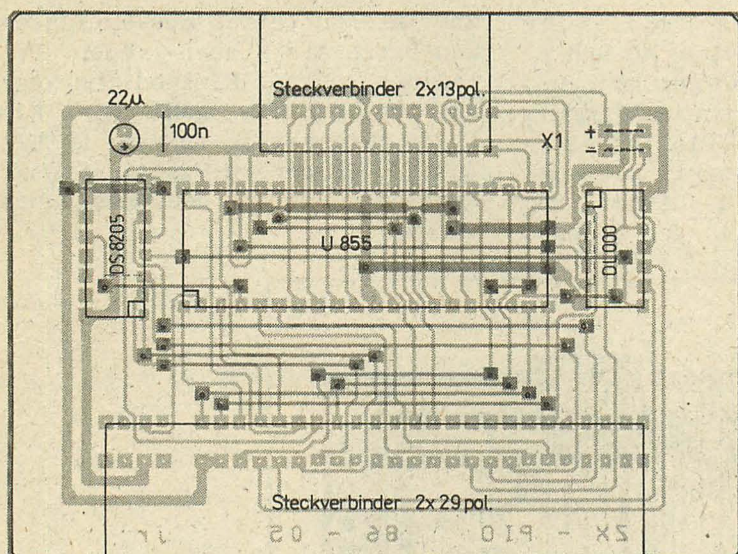


Bild 5.27 Bestückungsplan

- rückgesetzte Kanalmaskierungsregister,
- hochohmige Kanaldatenleitungen,
- inaktive READY-Leitungen (low-Zustand),
- PIO befindet sich in Betriebsart 1 (Byte-Eingabe),
- rückgesetzte Kanalausgaberegister,
- NICHT rückgesetztes Vektoradressenregister.

Die PIO verfügt aufgrund der Begrenzung der Anschlußzahl auf 40 über keinen RESET-Anschluß. Daher wurde eine andere Möglich-

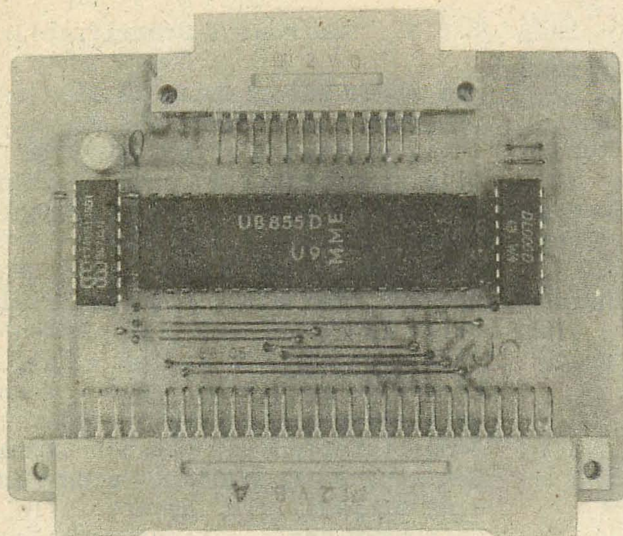


Bild 5.28 Musteraufbau der PIO-Karte

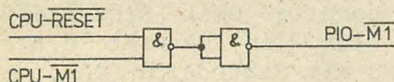


Bild 5.29 Schaltung zum Rücksetzen der PIO

keit vorgesehen, die PIO unabhängig vom Power-on-Reset in den Rücksetzzustand zu bringen. Diese besteht darin, bei inaktiven Signalen $\overline{RD}$ und $\overline{IORQ}$ mindestens zwei Taktperioden lang ein aktives $\overline{M1}$-Signal anzulegen. Das läßt sich mit einer zusätzlichen Gatterschaltung gemäß Bild 5.29 erreichen. In der Schaltung nach

Tabelle 5.5 Belegung des Anwendersteckverbinders des PIO-Moduls

Kontakt	A-Ebene	B-Ebene
1	IEI	IEO
2	$\overline{ARDY}$	$\overline{BRDY}$
3	$\overline{ASTB}$	$\overline{BSTB}$
4	A0	B0
5	A1	B1
6	A2	B2
7	A3	B3
8	A4	B4
9	A5	B5
10	A6	B6
11	A7	B7
12	Masse	+5 V
13	Masse	+5 V

Bild 5.24 sind dafür zwei weitere Gatter des Schaltkreises IS3 vorgesehen.
Zu beachten ist bei allen Anwendungen, daß die Portleitungen der PIO nur jeweils eine TTL-Last treiben können. Gegebenenfalls sind entsprechende Treiberschaltkreise vorzusehen.
Entsprechend der Beschaltung des Adreßdekoders ergibt sich für die PIO folgende Adreßzuordnung:

143 (8FH) Port A Daten
159 (9FH) Port B Daten

175 (AFH) Port A Steuerung
191 (BFH) Port B Steuerung

Die Überprüfung der Funktion der PIO-Karte kann mit Basic-IN- bzw. OUT-Befehlen an die entsprechenden Adressen vorgenommen werden. Die Belegung des 2 x 13-poligen Anwendersteckverbinders ist Tabelle 5.5 zu entnehmen.

5.6.3. *Centronics*-Schnittstelle

Mit einem entsprechenden Steuerprogramm ist es über die PIO relativ einfach möglich, eine Parallelschnittstelle für einen Drucker zu realisieren. Versuche haben gezeigt, daß eine PIO durchaus in der Lage ist, den Drucker über eine bis zu zwei Meter lange Leitung ohne Verwendung zusätzlicher Treiberschaltkreise zu bedienen. Eine sehr weit verbreitete Norm für solche Schnittstellen ist die der Firma *Centronics*. Es handelt sich um eine bitparallele, byteserielle, unidirektionale Schnittstelle, deren Signale im Bild 5.30 wiegegeben werden.
Für praktische Anwendungen genügen jedoch neben den Datenleitungen bereits zwei Steuerleitungen für den Betrieb eines Druckers mit *Centronics*-Schnitt-

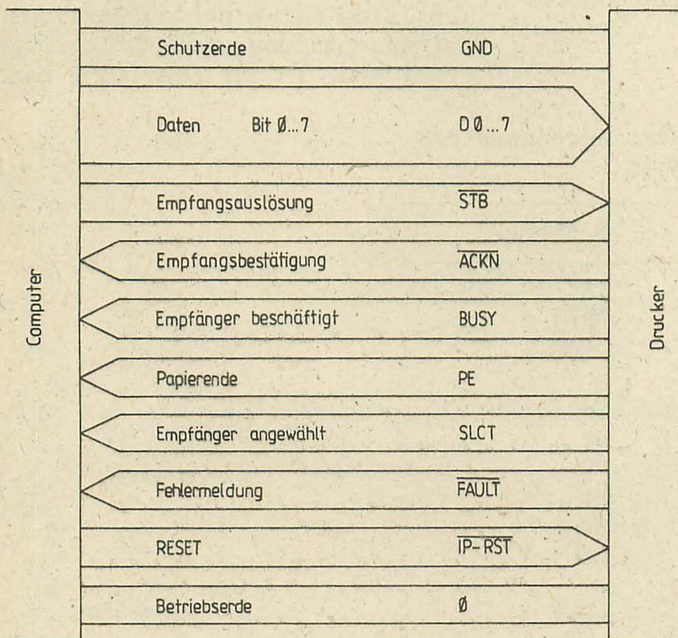


Bild 5.30 Signale der Centronics-Schnittstelle

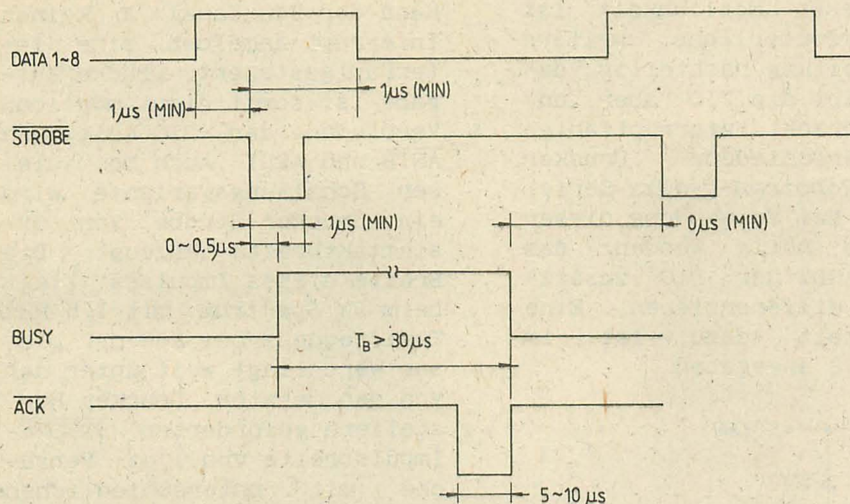


Bild 5.31 Signalspiel einer Minimal-Centronics-Schnittstelle

stelle. Es sind dies die Leitungen $\overline{\text{STROBE}}$ und $\overline{\text{BUSY}}$. Mit aktivem $\overline{\text{STROBE}}$-Signal teilt der Rechner dem Drucker mit, daß ein gültiges Byte auf den Datenleitungen ansteht, mit der $\overline{\text{BUSY}}$-Leitung zeigt der Drucker dem Rechner an, ob er zum Datenempfang bereit oder nicht bereit ist. Das Signalspiel einer solchen abgerüsteten Centronics-Schnittstelle ist dem Bild 5.31 zu entnehmen.

Die Erzeugung des Drucker-$\overline{\text{STROBE}}$-Signals wird von der PIO übernommen. Nachdem nämlich das Einschreiben eines Bytes in die PIO beendet ist (Ende des $\overline{\text{WR}}$-Impulses), wird nach der nächsten fallenden Flanke des Systemtaktes das READY-Signal der PIO in den aktiven Zustand (High-Pegel) gebracht. Da das Drucker-STROBE in der überwiegenden Zahl der Fälle low-aktiv ist, wird das READY-Signal der PIO über ein Gatter von IS3 invertiert. Dazu ist auf der Platine die Brücke X1(-), bei nichtinvertiertem READY sinngemäß X1(+) zu schließen. Diese Möglichkeit ist auf der Platine nur für den PIO-Port A vorgesehen, der in der vorgeschlagenen Variante als Datenport der *Centronics*-Schnittstelle arbeitet.

Bei Verwendung der angegebenen Treibersoftware ist die BUSY-Leitung des Druckers mit dem Bit2 des PIO-Ports B zu verbinden. Nachteilig an der beschriebenen Erzeugung des Drucker-STROBE ist zunächst die Tatsache, daß das Signal ARDY der PIO auch dann noch aktiv ist, wenn der Drucker die Daten längst übernommen hat. Um dies zu umgehen, gibt es verschiedene Möglichkeiten:

1. Verbindung des Drucker-Signals $\overline{\text{ACK}}$ mit $\overline{\text{ASTB}}$ der PIO. Dieses Drucker-Signal wird immer am Ende des $\overline{\text{BUSY}}$-Signals aktiviert. Es meldet das Ende der Datenübernahme. Bei der PIO löst die positive Flanke des Impulses die Deaktivierung des READY-Signals nach der nächsten fallenden Flanke des Systemtakts aus.

Bei dieser Möglichkeit ist die erforderliche weitere Steuerleitung nachteilig, dafür bleibt die PIO aber uneingeschränkt interruptfähig. Für verschiedene Drucker (u. a. *Robotron-K-63xx*-Serie) kann es bei Verwendung dieser Variante nötig werden, das Signal ARDY der PIO zusätzlich zu differenzieren. Eine Möglichkeit dazu ist im Bild 5. 32 angegeben.

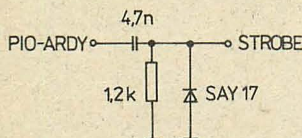


Bild 5. 32 Differenzierglied für das Drucker-Strobe-Signal

2. Verbindung von $\overline{ASTB}$ der PIO mit +5 V. In dieser Betriebsart erzeugt die PIO nach Einschreiben der Daten einen ARDY-Impuls der Länge einer Systemtaktperiode, der invertiert als Drucker-Strobe verwendet werden kann. Dabei kann der PIO-Kanal A keinen Interrupt anmelden, eine interruptgesteuerte Druckerausgabe ist somit nicht möglich.
3. Verbindung der PIO-Leitungen $\overline{ASTB}$ und ARDY. Auch bei dieser Schaltungsvariante wird ein Drucker-Strobe von Systemtaktbreite erzeugt. Die Breite dieses Impulses liegt beim *ZX Spectrum* mit 3, 5 MHz Taktfrequenz bei 286 ns. Dieser Wert liegt weit unter der von den meisten Drucker-Herstellern geforderten $\overline{STROBE}$-Impulsbreite von 1 µs. Versuche mit unterschiedlichen Druckern haben aber gezeigt, daß es hierbei keine Probleme mit der Datenübernahme gibt. Interrupts können auch hier nicht ausgelöst werden, da aufgrund der internen Schaltung der PIO sowohl der Zustand $\overline{M1}$ = 0 als auch $\overline{STROBE}$ = 0 eine Übernahme der Interruptanforderung in den internen Interruptspeicher verhindert.

Tabelle 5. 6 Centronics-Treiberprogramm

```
                       2 ;*************************
                       4 ;*** GP 550 - TREIBER ***
                       6 ;*************************
                       8 ;
F230                  10        ORG  62000
5C10                  20 STRMS  EQU  #5C10
4000                  30 VIM    EQU  #4000         ;ANFANG DISPLAYDATEI
5800                  40 ATTR   EQU  #5800
5C4F                  50 CHANS  EQU  #5C4F
0C10                  60 TOK    EQU  #0C10
F230 C350F2           70 JPT    JP   INIT          ;EINSPRUNGTABELLE
F233 C36CF2           80        JP   LST
F236 C39CF2           90        JP   SCPY
F239 1B543136        100 ING    DEFB 27,"T","1","6"
                                                   ;GRÖSSE ZEILENVORSCHUB
F23D 0D0A1B47        110 VTB    DEFB 13,10,27,"G","5","1","2"
     3132                                          ;512 GRAFIKDATEN
F244 1B36            120 INST   DEFB 27,"6"
```

```
F246 7E      130 MOV   LD   A,(HL)       ;STEUER-
F247 C5      140       PUSH BC           ;ZEICHEN
F248 CD81F2  150       CALL AGR          ;AUSGEBEN
F24B C1      160       POP  BC           ;ANZAHL
F24C 23      170       INC  HL           ;STEHT
F24D 10F7    180       DJNZ MOV          ;IM B-REGISTER
F24F C9      190       RET
F250 F3      200 INIT  DI                ;INITIALISIERUNG
F251 3E0F    210       LD   A,#0F
F253 D3AF    220       OUT  (#AF),A      ;PIO PORT A MODE 0
F255 3ECF    230       LD   A,#CF
F257 D3BF    240       OUT  (#BF),A      ;PIO PORT B MODE 3
F259 3E0F    250       LD   A,#0F
F25B D3BF    260       OUT  (#BF),A      ;UNTERE 4 BIT EINGANG
F25D 2A4F5C  270       LD   HL,(CHANS)   ;SYSTEMVARIABLE
F260 010F00  280       LD   BC,#000F     ;CHANS
F263 09      290       ADD  HL,BC        ;+ 15
F264 016CF2  300       LD   BC,LST       ;ZEIGT AUF
F267 71      310       LD   (HL),C       ;ANFANG
F268 23      320       INC  HL           ;DER
F269 70      330       LD   (HL),B       ;PRINTROUTINE
F26A FB      340       EI
F26B C9      350       RET
F26C F3      360 LST   DI                ;INTERRUPTSPERRE
F26D FE80    370       CP   #80          ;ZEICHEN DRUCKBAR ?
F26F 3805    380       JR   C,GRP        ;JA: AUSGEBEN
F271 D6A5    390       SUB  #A5          ;NEIN:
F273 C3100C  400       JP   TOK          ;TOKENBEHANDLUNG
F276 F3      410 GRP   DI
F277 FE0D    415       CP   13           ;ZEICHEN ENTER ?
F279 CCEEF2  420       CALL Z,ENTER      ;JA: ZEILENVORSCHUB
F27C CD81F2  425       CALL AGR          ;MIT AUSGEBEN
F27F FB      430       EI
F280 C9      440       RET
F281 CD89F2  450 AGR   CALL BUSY         ;DRUCKER BEREIT ?
F284 38FB    460       JR   C,AGR        ;NEIN: WARTEN
F286 D38F    470       OUT  (#8F),A      ;JA: ZEICHEN AUSGEBEN
F288 C9      480       RET
F289 47      490 BUSY  LD   B,A          ;REGISTER RETTEN
F28A 3E7F    500       LD   A,#7F
F28C DBFE    510       IN   A,(#FE)
F28E 1F      520       RRA
F28F 3803    530       JR   C,BM1
F291 FB      540       EI
F292 CF      550       RST  #08
F293 0C      560       DEFB #0C
F294 AF      570 BM1   XOR  A
F295 DB9F    580       IN   A,(#9F)      ;DRUCKERSTATUS LESEN
F297 1F      590       RRA               ;PIO-BIT B2
F298 1F      600       RRA               ;INS CARRY-FLAG
F299 1F      610       RRA               ;ROTIEREN
F29A 78      620       LD   A,B          ;REGISTER ZURÜCK
F29B C9      630       RET
```

```
F29C F3       640 SCPY   DI                ;BILDSCHIRM-KOPIE
F29D 2139F2   650        LD   HL,ING       ;STEUERWORTTABELLE
F2A0 060B     660        LD   B,11         ;11 STEUERWORTE
F2A2 CD46F2   670        CALL MOV          ;AUSGEBEN
F2A5 210040   680        LD   HL,VIM
F2A8 1E03     690        LD   E,3          ;3 BILDSCHIRMDRITTEL
F2AA 1621     700        LD   D,33         ;32 ZEICHEN JE ZEILE
F2AC 15       710 GM1    DEC  D            ;ZEILE FERTIG ?
F2AD CCE1F2   720        CALL Z,VOR        ;JA: NÄCHSTE ZEILE
F2B0 CDC9F2   730        CALL GR88
F2B3 2C       740        INC  L
F2B4 20F6     750        JR   NZ,GM1
F2B6 3E08     760        LD   A,8
F2B8 84       770        ADD  A,H
F2B9 67       780        LD   H,A
F2BA 1D       790        DEC  E            ;3 DRITTEL FERTIG ?
F2BB 20EF     800        JRNZ GM1          ;NEIN: WEITER
F2BD 0602     810        LD   B,2          ;HARDCOPY FERTIG
F2BF 2144F2   820        LD   HL,INST      ;EINEN ZEILENVORSCHUB
F2C2 CD46F2   830        CALL MOV          ;IM GRAFIKMODUS
F2C5 3E0A     835        LD   A,10         ;AUSGEBEN
F2C7 18AD     840        JR   GRP
F2C9 0E08     850 GR88   LD   C,8          ;8 X 8 PIXEL
F2CB E5       860 ROL1   PUSH HL
F2CC 0608     870        LD   B,8          ;PRO ZEICHENPOSITION
F2CE CB06     880 ROL2   RLC  (HL)
F2D0 1F       890        RRA
F2D1 24       900        INC  H
F2D2 10FA     910        DJNZ ROL2
F2D4 C5       920        PUSH BC
F2D5 CD81F2   930        CALL AGR          ;ZWEIMAL AUSGEBEN VER-
F2D8 CD81F2   935        CALL AGR          ;GRÖSSERT DAS BILD
F2DB C1       940        POP  BC
F2DC E1       950        POP  HL
F2DD 0D       960        DEC  C
F2DE 20EB     970        JR   NZ,ROL1
F2E0 C9       980        RET
F2E1 E5       990 VOR    PUSH HL           ;STEUERWORTTABELLE
F2E2 213DF2  1000        LD   HL,VTB       ;GRAFIKMODUS
F2E5 0607    1010        LD   B,7          ;7 STEUERWORTE
F2E7 CD46F2  1020        CALL MOV          ;AUSGEBEN
F2EA 1620    1030        LD   D,32         ;NÄCHSTE 32 POSITIONEN
F2EC E1      1040        POP  HL
F2ED C9      1050        RET
F2EE 3E0A    1060 ENTER  LD   A,10         ;ZEILENVORSCHUB
F2F0 CD81F2  1070        CALL AGR          ;AUSGEBEN
F2F3 3E0D    1080        LD   A,13         ;ENTER-CODE LADEN
F2F5 C9      1090        RET
F2F6         1100        END
```

Die vorgestellte *Centronics*-Treibersoftware (Tabelle 5.6) ist auf dem *ZX Spectrum* lauffähig. Sie akzeptiert die Funktionen LPRINT und LLIST direkt, die Funktion COPY muß über

Signal	Pin	Pin	Signal
$\overline{STB}$	1	19	Ø
DØ	2	20	Ø
D1	3	21	Ø
D2	4	22	Ø
D3	5	23	Ø
D4	6	24	Ø
D5	7	25	Ø
D6	8	26	Ø
D7	9	27	Ø
$\overline{ACKN}$	10	28	Ø
BUSY	11	29	Ø
PE	12	30	Ø
$\overline{SLCT}$	13	31	$\overline{IP-RST}$
	14	32	$\overline{FAULT}$
	15	33	Ø
Ø	16	34	
GND	17	35	
	18	36	

Bild 5.33 Genormte Belegung eines Centronics-Steckverbinders (Amphenol)

	A	B	C
1		PE	Ø
2		$\overline{STB}$	Ø
3		$\overline{ACKN}$	Ø
4		SLCT	Ø
5		DØ	Ø
6		D1	Ø
7		D2	Ø
8		D3	Ø
9	$\overline{FAULT}$	D4	Ø
1Ø	Ø	D5	$\overline{IP-RST}$
11	Ø	D6	BUSY
12	Ø	D7	Ø
13	GND		Ø

Bild 5.34 Anschlußbelegung der Drucker-Serie K 63xx

RANDOMIZE USR Startadresse + 6 aufgerufen werden. Darüber hinaus sind mit diesem Treiber auch Druckerausgaben aus anderen Programmen heraus möglich. Bedingung dafür ist, daß die Druckerausgaberoutine dieser Programme das Zeichen im A-Register des Prozessors über den aktuell geöffneten Kanal ausgibt. Nach dem Laden des Programms wird mit dem Befehl

RANDOMIZE USR Startadresse

die PIO initialisiert und dem Rechner mitgeteilt, wohin er die zu druckenden Daten schicken soll.

Für die schon erwähnte interruptgesteuerte Druckerausgabe wäre neben einem anderen Treiberprogramm die Verbindung des PIO-Anschlusses IEI mit +5 V oder dem Anschluß IEO des höher priorisierten Interface-Schaltkreises erforderlich. Natürlich können einfache Druckertreiber unter Nutzung der guten Stringverarbeitungsmöglichkeiten des *Spectrum* auch in Basic geschrieben werden. Dazu sei noch angemerkt, daß es viel Programmieraufwand und Speicherplatz spart, wenn häufig gebrauchte Steuersequenzen (Escape-Folgen) wie im folgenden Beispiel in eine Stringvariable übernommen werden:

```
10 LET G$=CHR$(27)+"K"
```

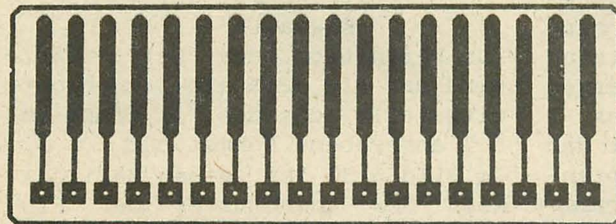


Bild 5.35 Behelfslösung für Amphenol-Steckverbinder (Einzelplatine)

```
+CHR$(100)+CHR$(0)
20 LPRINT G$
```

Die Bilder 5.33 und 5.34 zeigen die Anschlußbelegungen der verwendeten Drucker. Während der Steckverbinder für den *K 63xx* sicher leicht zu beschaffen ist, gibt es bei den *Amphenol*-Steckverbindern manchmal Schwierigkeiten. Im Bild 5.35 ist dafür eine Behelfslösung angegeben. Diese Leiterplatte wird zweimal angefertigt; die Rückseiten sind mit einer 0,5 mm dicken Blechzwischenlage nach Bild 5.36 zusammenzukleben. Dazu kann z. B. *Helapox schnellhärtend* verwendet werden. Es empfiehlt sich sehr, diesen Ersatzstecker zu verzinnen, da es anderenfalls oft zu Kontaktschwierigkeiten kommt.

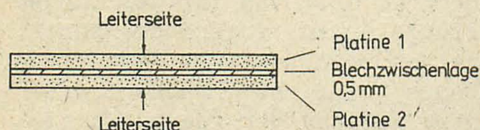


Bild 5.36 Aufbau der Amphenol-Behelfslösung

5.6.4. Programmierung der PIO *U 855* im Interruptbetrieb

Die Programmierung von Interruptanforderungen wird beim *ZX Spectrum* durch das ULA sehr erschwert. Dieses löst hardwaremäßig alle 20 ms einen Interrupt aus, in dem u. a. die Tastatur abgefragt wird. Der Prozessor läuft dazu im Interruptmodus 1 (IM1). In diesem Modus führt der Prozessor bei jeder Interruptanforderung einen Unterprogrammaufruf auf Adresse 38H aus. Erst der viel leistungsfähigere IM2 erlaubt es aber, das Programm in Abhängigkeit von der peripheren Interruptquelle zu verzweigen. Leider ist dafür das ULA nicht ausgelegt. Es ist daher programmtechnisch einfacher, auf den ULA-Interrupts aufzubauen und mit der Tastaturabfrage z. B. auch einen PIO-Port einzulesen. Da das dann ebenfalls alle 20 ms abläuft, sind auch relativ zeitkritische Aufgaben lösbar.

5.6.5. Parallelschnittstelle mit der PIO *8255*

Der Schaltkreis *8255* entstammt dem Prozessorsystem *8080*, ist aber auch im System *U 880* einsetzbar. Gegenüber der PIO *U 855* weist er 24 Ein-/Ausgänge auf und ist einfacher, aber nicht so universell programmierbar. Die 24 Ein-/Ausgänge sind in 3 Ports zu je 8 bit unterteilt. Der Port C kann in 2 Gruppen zu je 4 bit dem Port B (C0 bis C3) und dem Port A (C4 bis C7) zugeordnet werden, womit die PIO *8255* eine 2-Port-Struktur zu je 12 bit aufweist. Die PIO *8255* läßt sich in 3 Betriebsarten (Moden) betreiben:

- Mode 0 - Standard-Ein-/Ausgabe ohne Quittungssignale,
- Mode 1 - getaktete Ein-/Ausgabe mit Quittungssignalen,
- Mode 2 - bidirektionaler Betrieb mit Quittungssignalen.

Zur Definition der Betriebsart wird der PIO *8255* ein Steuerwort mitgeteilt, das das in Tabelle 5.7 angegebene Format hat.
Im Bild 5.37 ist eine Möglichkeit des Anschlusses der PIO *8255* an den *ZX Spectrum* gezeigt. Diese Schaltung benutzt die gleichen Adressen wie die Schal-

Tabelle 5.7 Steuerwort-Aufbau des PIO-Schaltkreises *8255*

Bit	0/1	Bedeutung
Gruppe B		
D0	0	Port C untere 4 Bit Ausgang
	1	Port C untere 4 Bit Eingang
D1	0	Port B Ausgang
	1	Port B Eingang
D2	0	Betriebsart 0
	1	Betriebsart 1
Gruppe A		
D3	0	Port C obere 4 Bit Ausgang
	1	Port C obere 4 Bit Eingang
D5/D6	0 0	Betriebsart 0
	0 1	Betriebsart 1
	1 x	Betriebsart 2
D7	0	Betriebsart definieren
	1	Einzelbit setzen/löschen

tung mit der PIO *U 855*, da in Kleincomputersystemen vermutlich eine PIO ausreicht. Die PIO *8255* hat einen H-aktiven Reset-An-

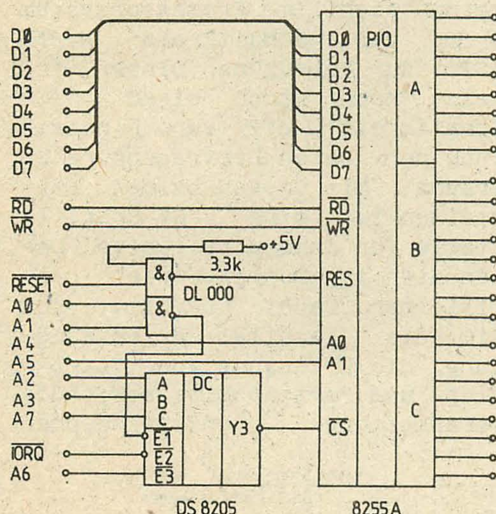


Bild 5.37 Anschluß der PIO 8255 an den ZX Spectrum

schluß, so daß die Reset-Leitung des Rechners invertiert werden muß, falls nicht nur vom Power-on-Reset Gebrauch gemacht wird. Nach einem Reset sind alle internen Register zurückgesetzt, die Ports A, B und C sind als Eingänge definiert. Die Anschlußbelegung der PIO *8255* ist dem Bild 5.38 zu entnehmen.

5.7. Serielle Schnittstellen

Im Gegensatz zu einer Parallel-Schnittstelle, bei der ganze Bytes übergeben werden, erfolgt bei der seriellen Schnittstelle der Datentransport Bit für Bit als Folge von Spannungsimpulsen einer bestimmten Dauer. Ein sehr weit verbreiteter Standard für serielle Schnittstellen ist die Norm V.24 bzw. RS 232 C. Hierbei werden Pakete von 5 bis 8 Daten-

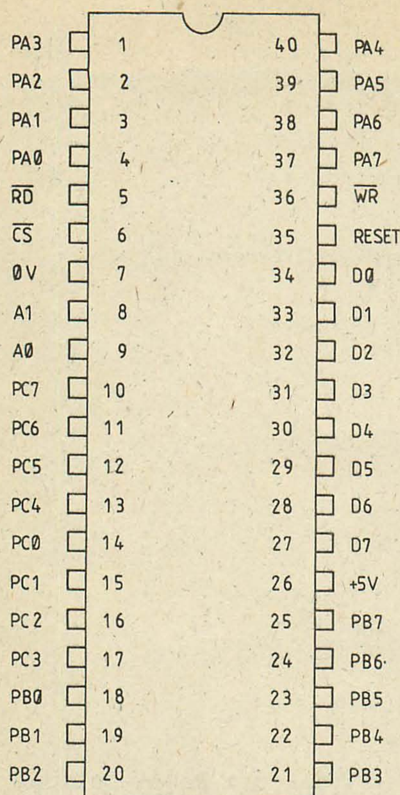


Bild 5.38 Anschlußbelegung der PIO 8255

bits übertragen, die von einem Startbit und ein bis zwei Stopbits eingerahmt sind. Zusätzlich kann vor dem Stopbit ein Paritätsbit vereinbart werden, das die Anzahl der High-Bits im Datenwort auf einen geraden oder ungeraden Wert ergänzt. Das Prinzip einer V.24-Datenübertragung ist im Bild 5.39 dargestellt. Daraus geht hervor, daß das Startbit grundsätzlich High- und das Stopbit grundsätzlich Low-Pegel hat.

Zur Erhöhung der Störungsfestigkeit wird logisch Eins (gesetztes Bit) nicht durch +5 V, sondern durch eine Spannung von -3 bis -12 V realisiert, logisch Null dementsprechend mit +3 bis +12 V. Für eine solche Schnittstelle sind also neben der Signalaufbereitung auch Pegelwandler erforderlich, die als integrierte Schaltkreise zur Verfügung stehen oder entsprechend Bild 5.40 diskret aufgebaut werden können. Die V.24-Schnittstelle arbeitet mit verschiedenen, genormten Datenübertragungsraten, die in Baud (bit pro Sekunde) angegeben werden. Die am häufigsten verwendeten Übertragungsraten sind 75, 110, 135, 150, 300, 600, 1200, 2400, 4800, 9600 und 19200 Baud. Über eine V.24-Schnittstelle ist Duplexbetrieb möglich, d. h., der Computer kann gleichzeitig Daten senden und empfangen. Dabei können Sender und Empfänger mit unterschiedlichen Datenübertragungsraten betrieben werden.

Für die Realisierung einer seriellen Schnittstelle im allgemeinen steht im Prozessorsystem *U 880* der Schaltkreis *U 856* (SIO) zur Verfügung. Dieser wird meist noch durch einen CTC-Schaltkreis *U 857* zur Einstellung der Datenübertragungsraten ergänzt. Mit diesen beiden Bausteinen ist eine V.24-Schnittstelle für nahezu beliebige Datenraten im Duplexbetrieb realisierbar. Dabei übernimmt die SIO die Parallel-Serien-Wandlung, die Erzeugung von Start-, Stop- und Paritätsbits sowie die Verarbeitung entsprechender

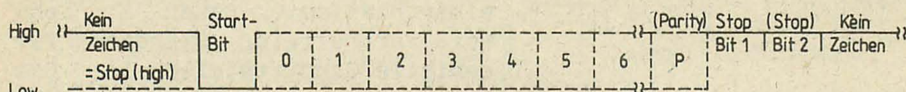


Bild 5.39 Prinzip der V.24-Übertragung

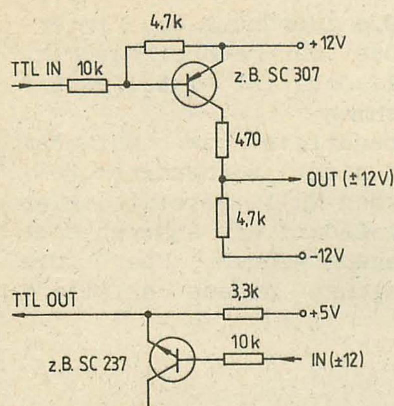


Bild 5.40 Pegelwandler für V. 24-Schnittstellen

Steuersignale. Schaltungsbeispiele für solche Schnittstellen sind in der Literatur bereits mehrfach veröffentlicht worden [2], [8], [9].
Beide Schaltkreise belegen allerdings jeweils 4 E/A-Adressen. Das läßt einen solchen Aufbau für den *ZX Spectrum* mit seinem ohnehin stark eingeschränkten Adreßraum ungeeignet erscheinen, zumal der vermutliche Hauptanwendungsfall, der Betrieb eines Druckers mit V. 24-Schnittstelle, einen solchen Aufwand nicht rechtfertigt. Über eine PIO oder einen anderen E/A-Baustein ist es jedoch möglich, mit geeigneter Software eine V. 24-Schnittstelle zu simulieren. Das soll am Beispiel des Anschlusses der elektronischen Schreibmaschine *Erika S 3004* an den *ZX Spectrum* näher erläutert werden.

5.7.1. Anschluß der *Erika S 3004*

Die Typenrad-Schreibmaschine *S 3004* enthält einen eigenen Mikrorechner, der sämtliche internen Abläufe steuert. Es wird der Einchiprechner *UB 8840* angewendet, dessen Programm auf einem externen EPROM *U 2732* (4K x 8 bit) abgelegt ist. Über Aufbau, Funktion, Befehlssatz und Programmierung dieses Einchiprechners kann u. a. in [10] nachgelesen werden.
Zur S 3004 werden als Option verschiedene Interface-Kassetten angeboten, die aber zum Zeitpunkt der Manuskripterarbeitung noch nicht zur Verfügung standen. Geplant sind V. 24-, Centronics- und *Commodore*-Schnittstellen. Für den Anschluß dieser Interface-Kassetten dient ein an der rechten Seite der Maschine befindlicher 2 x 13poliger EGS-Steckverbinder (Bild 5.41). Von den 26 Leitungen sind allerdings nur ganze 6 wirklich beschaltet. und positive Betriebsspannung (+5 V) sind auf diesen Steckverbinder die Portleitungen P30, P31, P36 und P37 des *UB 8840* geführt. Die Leitungen P30 und P31 sind serielle Eingänge, die Leitungen P36 und P37 serielle Ausgänge des Einchiprechners.
Von Bedeutung für die Anwendung der *S 3004* als Drucker sind die Leitungen P30 (Datenleitung) und P36 (BUSY-Signal). Die Schnittstelle arbeitet mit einer Datenübertragungsrate von 1200 Baud, einem Startbit, 8 Datenbits und zwei Stopbits ohne Parität.
Leider ist die direkte Ansteuerung der Schreibmaschine mit ASCII-codierten Zeichen nicht möglich, da intern ein Code verwendet wird, der der Position des Zeichens auf dem Typenrad entspricht. Die Umcodierung könnte hardwaremäßig, etwa über einen EPROM, vorgenommen werden, was aber den Aufwand wieder erhöht. Deshalb wird vom Treiberprogramm eine Umcodierung über eine Zuordnungstabelle durchge-

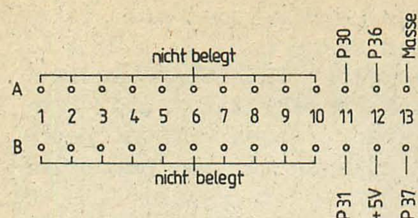


Bild 5.41 Anschlußbelegung des Interface-Steckverbinders der Erika S 3004

führt. Die Zuordnung des internen Codes der Schreibmaschine zum ASCII-Code ist Tabelle 5.8 zu entnehmen.
Nicht benötigte bzw. auf dem Typenrad nicht vorhandene Zeichen wurden Null gesetzt. Hier kann bei Bedarf ein anderer Code eingetragen werden. Über die ASCII-Zeichen hinaus existiert

Tabelle 5.8 Interne Codes der Schreibmaschine *Erika S 3004*

ASCII dez.	hex.	Zeichen	*S 3004* dez.	hex.
0	00		0	00
1	01		0	00
2	02		0	00
3	03		0	00
4	04		0	00
5	05		0	00
6	06		0	00
7	07		0	00
8	08		0	00
9	09		0	00
10	0A		0	00
11	0B		0	00
12	0C		0	00
13	0D	ENTER	11	77
14	0E		0	00
15	0F		0	00
16	10		0	00
17	11		0	00
18	12		0	00
19	13		0	00
20	14		0	00
21	15		0	00
22	16		0	00
23	17		0	00
24	18		0	00
25	19		0	00
26	1A		0	00
27	1B		0	00
28	1C		0	00
29	1D		0	00
30	1E		0	00
31	1F		0	00

ASCII dez.	hex.	Zeichen	*S 3004* dez.	hex.
32	20	SPACE	113	71
33	21	!	66	42
34	22	"	67	43
35	23	#	65	41
36	24	$	72	48
37	25	%	4	04
38	26	&	2	02
39	27	'	23	17
40	28	(	30	1E
41	29	)	31	1F
42	2A	*	27	1B
43	2B	+	37	25
44	2C	,	100	64
45	2D	-	98	62
46	2E	.	99	63
47	2F	/	64	40
48	30	0	0	0D
49	31	1	17	11
50	32	2	16	10
51	33	3	15	0F
52	34	4	14	0E
53	35	5	12	0C
54	36	6	11	0B
55	37	7	10	0A
56	38	8	9	09
57	39	9	8	08
58	3A	:	19	13
59	3B	;	59	3B
60	3C	<	0	00
61	3D	=	46	2E
62	3E	>	0	00
63	3F	?	53	35
64	40	§	0	00
65	41	A	48	30
66	42	B	24	18
67	43	C	32	20
68	44	D	20	14
69	45	E	52	34
70	46	F	62	3E
71	47	G	28	1C
72	48	H	18	12
73	49	I	33	21
74	4A	J	50	32
75	4B	K	36	24
76	4C	L	44	2C
77	4D	M	22	16

ASCII dez.	hex.	Zeichen	*S 3004* dez.	hex.
78	4E	N	42	2A
79	4F	O	30	1E
80	50	P	47	2F
81	51	Q	26	1A
82	52	R	54	36
83	53	S	51	33
84	54	T	55	37
85	55	U	40	28
86	56	V	34	22
87	57	W	45	2D
88	58	X	38	26
89	59	Y	49	31
90	5A	Z	56	38
91	5B	Ä	63	3F
92	5C	Ö	60	3C
93	5D	Ü	58	3A
94	5E	^	0	00
95	5F	_	1	01
96	60	`	6	06
97	61	a	97	61
98	62	b	78	4E
99	63	c	87	57
100	64	d	83	53
101	65	e	90	5A
102	66	f	73	49
103	67	g	96	60
104	68	h	85	55
105	69	i	5	05
106	6A	j	75	4B
107	6B	k	80	50
108	6C	l	77	4D
109	6D	m	74	4A
110	6E	n	92	5C
111	6F	o	94	5E
112	70	p	91	5B
113	71	q	82	52
114	72	r	89	59
115	73	s	88	58
116	74	t	86	56
117	75	u	93	5D
118	76	v	79	4F
119	77	w	76	4C
120	78	x	95	5F
121	79	y	81	51
122	7A	z	84	54
123	7B	ä	101	65

ASCII dez.	hex.	Zeichen	*S 3004* dez.	hex.
124	7C	ö	102	66
125	7D	ü	103	67
126	7E	ß	71	47
127	7F		0	00

eine Anzahl Steuerwörter (Escape-Folgen), deren Bedeutung aus der Tabelle 5.9 ersichtlich ist. Beim Senden der als nicht belegt gekennzeichneten Codes reagiert die Schreibmaschine zwar auch, allerdings konnten dabei keine sinnvollen Funktionen nachgewiesen werden. Wahrscheinlich ist, daß diese Codes Testzwecken beim Hersteller vorbehalten sind. Für den Betrieb der *S 3004* an der beschriebenen V.24-Schnittstelle sind keine Pegelwandler erforderlich, da sowohl die PIO als auch der Einchiprechner lediglich TTL-Pegel verarbeiten. Somit beschränkt sich das Interface neben der PIO-Karte aus Abschnitt 5.6.2. auf ein dreiadriges Kabel, das Computer und Schreibmaschine gemäß Tabelle 5.10 verbindet.

An dieser Stelle sei noch angemerkt, daß die Portleitungen des Einchiprechners der Schreibmaschine ungepuffert auf den Interface-Steckverbinder geführt sind. Unachtsamkeiten (falsche Pegel, vertauschte Leitungen, Verbindung einer Portleitung mit Masse oder +5 V usw.) können hier relativ große (und teure) Schäden verursachen. Es wird deshalb empfohlen, trotz des etwas größeren Aufwands nicht

Tabelle 5.9 Steuercodes des *Erika S 3004*

Code *S 3004* dez.	hex.	Funktion
113	71	Leerzeichen (Space)
114	72	Rückschritt (Backspace)
115	73	Halbschritt rechts
116	74	Halbschritt links
117	75	Halbschritt hoch
118	76	Halbschritt runter
119	77	Wagenrücklauf/Zeilenschaltung (CR/LF)
120	78	Wagenrücklauf (CR)
121	79	Tabulator
122	7A	Tabulator auf aktuelle Position setzen
123	7B	Tabulator auf aktueller Position löschen
124	7C	Alle Tabulatoren löschen
125	7D	Tabulator-Raster setzen (auf 15, 25, 35, 45, 55, 65)
126	7E	linken Rand auf aktuelle Position setzen
127	7F	rechten Rand auf aktuelle Position setzen
128	80	Rand lösen

Code S 3004 dez.	hex.	Funktion
129	81	Mikrostep hoch
130	82	Mikrostep runter
131	83	Papiereinzug
132	84	Schreiben 1zeilig
133	85	Schreiben 1,5zeilig
134	86	Schreiben 2zeilig
135	87	10 Zeichen pro Zoll
136	88	12 Zeichen pro Zoll
137	89	15 Zeichen pro Zoll
138	8A	nicht belegt
139	8B	Druckstatus - Druck aller folgenden Zeichen
140	8C	Korrekturstatus - Korrektur aller folgenden Zeichen
141	8D	Vorwärtsdruck
142	8E	Rückwärtsdruck
143	8F	Ränder ignorieren
144	90	Ränder berücksichtigen
145	91	Druckwerk aus, Zeichenausgabe an Schnittstelle
146	92	Druckwerk ein
147	93	nicht belegt
148	94	nicht belegt
149	95	Schreibmaschine in Grundstellung
150	96	nicht belegt
151	97	nicht belegt
152	98	nicht belegt
153	99	nicht belegt
154	9A	nicht belegt
155	9B	Freigabe Dauerfunktion
156	9C	Sperre Dauerfunktion
157	9D	nicht belegt
158	9E	nicht belegt
159	9F	nicht belegt
160	A0	ständig Dauerfunktion
161	A1	Laden der Übertragungsrate der Interfacebox
162	A2	nicht belegt
.	.	.
.	.	.
169	A9	nicht belegt
170	AA	Hupe (Folgebyte bestimmt Dauer)
171	AB	nicht belegt
.	.	.
.	.	.
175	AF	nicht belegt
176	B0	frei
.	.	.
.	.	.
255	FF	frei

Tabelle 5.10 Beschaltung des Interface-Kabels

PIO-Portleitung	Steckverbinder	Signal *S 3004*	Steckverbinder
A6	A10	RXD	A11
A7	A11	RTS	A12
Masse	A12/13	GND	A13

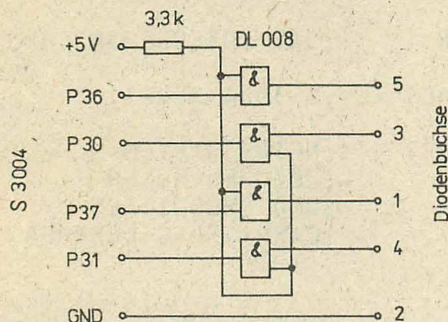


Bild 5.42 S 3004-Pufferstufe mit DL 008

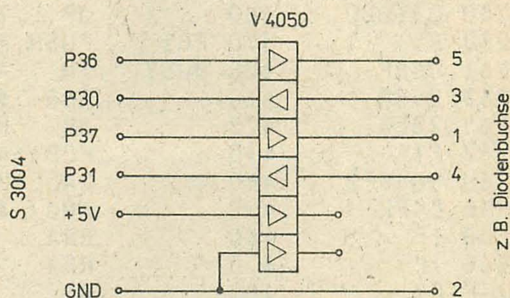


Bild 5.43 S 3004-Pufferstufe mit V 4050

Tabelle 5.11 S 3004-Treiberprogramm

```
                  10 ;*************************
                  20 ;*** S 3004 - TREIBER ***
                  30 ;*************************
                  40 ;
F230              50          ORG  62000        ;ANFANGSADRESSE
008F              60 PAD      EQU  #8F          ;PIO PORT A DATEN
00AF              70 PAC      EQU  #AF          ;PIO PORT A STEUERUNG
5C4F              80 CHANS    EQU  #5C4F        ;ZEIGER DRUCKROUTINE
0C10              90 TOKEN    EQU  #0C10        ;TOKENBEHANDLUNG
F230 F3          100 INIT     DI
F231 F5          110          PUSH AF
F232 C5          120          PUSH BC
F233 E5          130          PUSH HL
F234 3ECF        140          LD   A,#CF
F236 D3AF        150          OUT  (PAC),A      ;PORT A MODE 3
F238 3E80        160          LD   A,#80
F23A D3AF        170          OUT  (PAC),A      ;BIT 7 EINGANG
F23C 3E03        180          LD   A,3
F23E D3AF        190          OUT  (PAC),A      ;INTERRUPT AUS
F240 3E40        200          LD   A,#40
F242 D38F        210          OUT  (PAD),A      ;DATENAUSGANG HIGH
F244 2A4F5C      220          LD   HL,(CHANS)
F247 010F00      230          LD   BC,#000F     ;IN SYSTEMVARIABLE
F24A 09          240          ADD  HL,BC        ;CHANS + 15
F24B 0156F2      250          LD   BC,PRINT     ;ANFANGSADRESSE
```

```
F24E 71        260        LD   (HL),C      ;DER PRINT-
F24F 23        270        INC  HL          ;ROUTINE
F250 70        280        LD   (HL),B      ;LADEN
F251 E1        290        POP  HL
F252 C1        300        POP  BC
F253 F1        310        POP  AF
F254 FB        320        EI
F255 C9        330        RET
F256 F3        340 PRINT  DI
F257 FE80      350        CP   #80         ;ZEICHEN GRÖSSER 128 ?
F259 3805      360        JR   C,PRI       ;NEIN: DRUCKEN
F25B D6A5      370        SUB  #A5
F25D C3100C    380        JP   TOKEN       ;JA: TOKENBEHANDLUNG
F260 F5        390 PRI    PUSH AF
F261 DB8F      400 BUSY   IN   A,(PAD)     ;S 3004 BEREIT ?
F263 E680      410        AND  #80
F265 20FA      420        JR   NZ,BUSY     ;NEIN: WARTEN
F267 F1        430        POP  AF          ;ZEICHEN HOLEN
F268 CD68F2    440        CALL UMCOD       ;UND UMCODIEREN
F26B E6FF      450        AND  #FF         ;CARRY-FLAG LÖSCHEN
F26D 1F        460        RRA
F26E 1F        470        RRA
F26F F3        480        DI
F270 0609      490        LD   B,9         ;START- UND 8 DATENBIT
F272 D38F      500 AUSG   OUT  (PAD),A     ;AUSGEBEN
F274 1F        510        RRA
F275 CD75F2    520        CALL ZEIT        ;MIT 1200 BAUD
F278 10F8      530        DJNZ AUSG        ;BIS B = 0
F27A 3E40      540        LD   A,#40       ;DATENAUSGANG = 0
F27C D38F      550        OUT  (PAD),A     ;SETZEN
F27E CD86F2    560        CALL ZEIT        ;STOPBITS
F281 CD86F2    570        CALL ZEIT        ;AUSGEBEN
F284 FB        580        EI
F285 C9        590        RET
F286 C5        600 ZEIT   PUSH BC
F287 06D9      610        LD   B,217       ;ZEITSCHLEIFE
F289 10FE      620 ZEIT1  DJNZ ZEIT1       ;FÜR
F28B C1        630        POP  BC          ;1200 BAUD
F28C C9        640        RET
F28D C5        650 UMCOD  PUSH BC
F28E E5        660        PUSH HL
F28F 219AF2    670        LD   HL,COTAB    ;TABELLENANFANG
F292 4F        680        LDP  C,A         ;ALTER CODE IN C
F293 0600      690        LD   B,0
F295 09        700        ADD  HL,BC
F296 7E        710        LD   A,(HL)      ;NEUER CODE NACH A
F297 F1        720        POP  HL
F298 C1        730        POP  BC
F299 C9        740        RET
F29A 000000    750 COTAB  DEFB #00,#00,#00;CODETABELLE
F29D 000000    760        DEFB #00,#00,#00
F2A0 790072    770        DEFB #79,#00,#72
F2A3 000000    780        DEFB #00,#00,#00
```

```
F2A6 007700    790    DEFB #00,#77,#00
F2A9 000000    800    DEFB #00,#00,#00
F2AC 000000    810    DEFB #00,#00,#00
F2AF 000000    820    DEFB #00,#00,#00
F2B2 000000    830    DEFB #00,#00,#00
F2B5 000000    840    DEFB #00,#00,#00
F2B8 000071    850    DEFB #00,#00,#71
F2BB 424341    860    DEFB #42,#43,#41
F2BE 480402    870    DEFB #48,#04,#02
F2C1 171D1F    880    DEFB #17,#1D,#1F
F2C4 1B2564    890    DEFB #1B,#25,#64
F2C7 626340    900    DEFB #62,#63,#40
F2CA 0D1110    910    DEFB #0D,#11,#10
F2CD 0F0E0C    920    DEFB #0F,#0E,#0C
F2D0 0B0A09    930    DEFB #0B,#0A,#09
F2D3 08133B    940    DEFB #08,#13,#3B
F2D6 1D2E1F    950    DEFB #1D,#2E,#1F
F2D9 350030    960    DEFB #35,#00,#30
F2DC 182014    970    DEFB #18,#20,#14
F2DF 343E1C    980    DEFB #34,#3E,#1C
F2E2 122132    990    DEFB #12,#21,#32
F2E5 242C16   1000    DEFB #24,#2C,#16
F2E8 2A1E2F   1010    DEFB #2A,#1E,#2F
F2EB 1A3633   1020    DEFB #1A,#36,#33
F2EE 372822   1030    DEFB #37,#28,#22
F2F1 2D2631   1040    DEFB #2D,#26,#31
F2F4 383F3C   1050    DEFB #38,#3F,#3C
F2F7 3A6401   1060    DEFB #3A,#64,#01
F2FA 57614E   1070    DEFB #57,#61,#4E
F2FD 57535A   1080    DEFB #57,#53,#5A
F300 496055   1090    DEFB #49,#60,#55
F303 054B50   1100    DEFB #05,#4B,#50
F306 4D4A5C   1110    DEFB #4D,#4A,#5C
F309 5E5B52   1120    DEFB #5E,#5B,#52
F30C 595856   1130    DEFB #59,#58,#56
F30F 5D4F4C   1140    DEFB #5D,#4F,#4C
F312 5F5154   1150    DEFB #5F,#51,#54
F315 656667   1160    DEFB #65,#66,#67
F318 47       1170    DEFB #47
F319          1180    END
```

auf eine Pufferstufe zu verzichten. Geeignet sind u. a. die Schaltkreise *U 4050* (enthält 6 CMOS-Treiber) und *DL 008* (4 AND-Gatter), wie den Bildern 5.42 und 5.43 zu entnehmen ist. Bei Verwendung des CMOS-Treibers sind eventuell Pull-up-Widerstände erforderlich, wenn er mit TTL-Pegel angesteuert wird. Eine sehr elegante Lösung ist auch der Einsatz von Optokopplern.

Tabelle 5.11 enthält das kommentierte Listing des Treiberprogramms. Es ist lauffähig auf der 48-kByte-Version des *ZX Spectrum* und kann auf beliebige Adressen im oberen 32-kByte-Speicherbereich geladen werden. Das Programm INIT muß nach dem Laden einmal durchlaufen werden. In der unteren 32-kByte-Hälfte ist das Programm nicht ohne weiteres lauffähig, da durch die mit dem

Programmlauf nicht synchronisierten ULA-Zugriffe (Bildaufbau) die Zeitkonstanten verändert werden.

5.7.2. Fernschreiber-Anschluß

Ein anderer serieller, sehr störsicherer Standard ist die Norm RS 232/TTY. Hier werden die logischen Zustände durch das Fließen oder Fehlen eines Stroms von meist 20 mA dargestellt. Mit einer solchen Schnittstelle sind z. B. Fernschreiber ausgestattet, die zuweilen als Altgeräte sehr preiswert erhältlich sind. Sie lassen sich als einfache (wenn auch sehr laute) Ausgabemöglichkeit für Kleincomputer einsetzen, wenngleich Schriftbild und Zeichenumfang nur geringen Ansprüchen gerecht werden. Zur Kopplung eines Fernschreibers mit dem *ZX Spectrum* kann das im Bild 5.44 dargestellte Interface verwendet werden. Um einen Strom von 20 mA durch den Empfangsmagneten des Fernschreibers zu treiben, ist eine Spannung von etwa 40 V erforderlich. Diese Spannung kann sinnvoll nur durch ein separates Netzteil erzeugt werden. Um bei eventuellen Fehlern in diesem Netzteil den Computer nicht zu gefährden, wurden Nieder- und Hochspannungsseite des Interfaces durch einen Optokoppler galvanisch getrennt. Die Ansteuerimpulse werden durch den *Spectrum* an dessen Tonband-Anschluß bereitgestellt (MIC- bzw. EAR-Buchse). Mit einem Operationsverstärker (IC1) werden die Im-

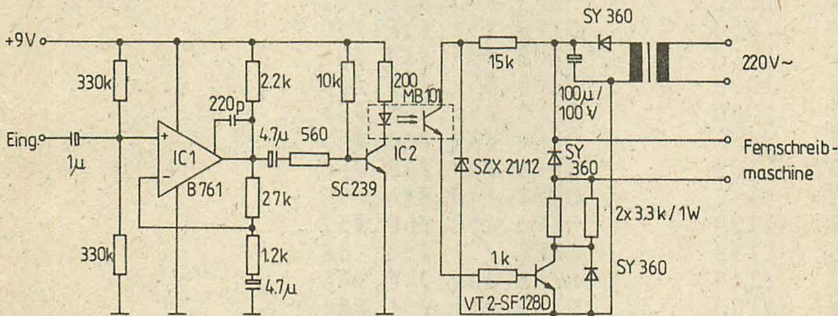


Bild 5.44 RS 232/TTY-Interface

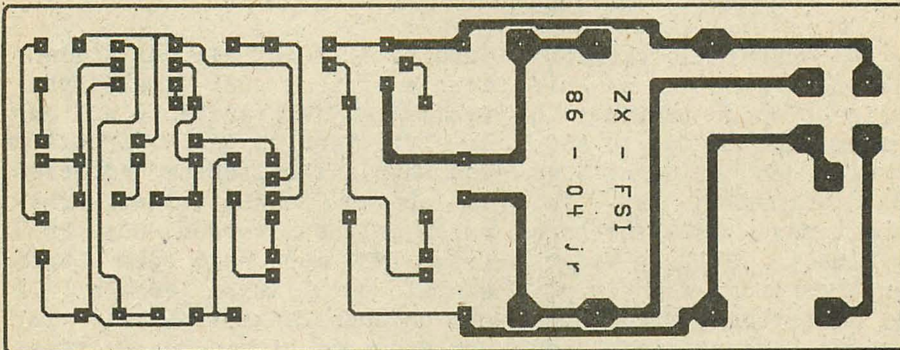


Bild 5.45 Layout für das RS 232/TTY-Interface

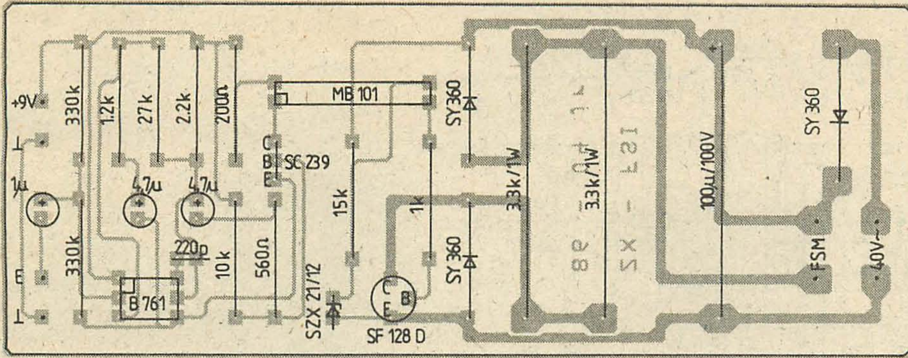


Bild 5.46 Bestückungsplan

pulse auf etwa 9 V verstärkt und über einen Schalttransistor dem Optokoppler IC2 zugeführt. Dieser schaltet auf der Hochspannungsseite den Transistor VT2.

Die Sekundärspannung des Netztransformators ist so zu bemessen, daß durch den Empfangsmagneten des Fernscheibers 20 mA fließen. Diese Spannung kann zunächst experimentell mit einem Stelltransformator ermittelt werden. Primärseitig ist eine Sicherung von etwa 35 mA vorzusehen. Da die Schaltung Netzpotential führt, ist ein berührungssicherer Aufbau, gegebenenfalls unter Hinzuziehung eines Fachmanns, erforderlich! Zur Ansteuerung des Fernschreiber-Interfaces kann ein Programm verwendet werden, das ähnlich wie das Steuerprogramm für die elektronische Schreibmaschine arbeitet. Natürlich sind dann die Ausgabeadresse, die Baudrate und die Codiertabelle entsprechend zu ändern. Layout und Bestückungsplan für das Fernschreiber-Interface sind in den Bildern 5.45 und 5.46 dargestellt, den Musteraufbau zeigt Bild 5.47.

5.8. Pegelanpassung für Kassettenrecorder

Unter den Massenspeichern für Heim- und Kleincomputer nehmen zumindest zur Zeit Magnetbandkassetten eine dominierende Stellung ein. Trotz der zunehmenden Verfügbarkeit schneller und sicherer Speichermedien auch für diese Geräteklasse lohnt es sich deshalb, die Übertragung vom bzw. zum Kassettenrecorder zu verbessern. Dadurch kann eine höhere Datensicherheit oder eine

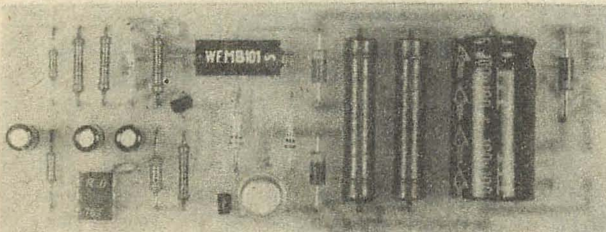


Bild 5.47 Musteraufbau

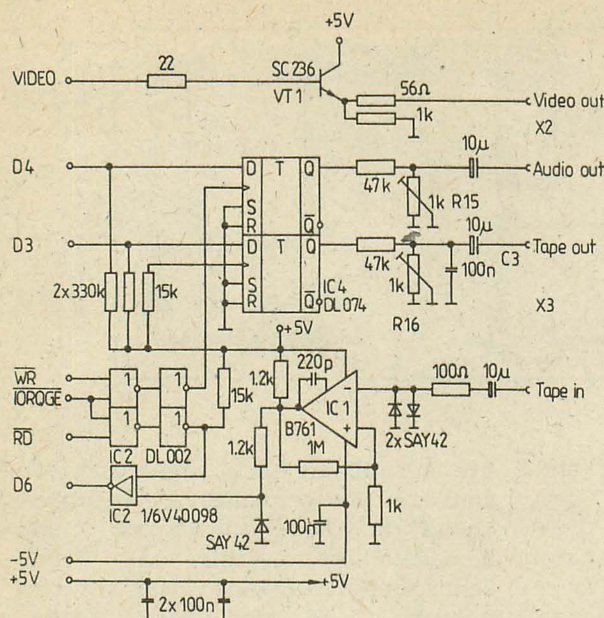


Bild 5.48 Schaltung der SAVE/LOAD-Pegelanpassung

schnellere Aufzeichnung erreicht werden. Die Pegelanpassung ermöglicht die Verwendung von Aufzeichnungsgeräten mit Diodenbuchse an (pegelmäßig) nicht dafür geeigneten Computern. Über das Interface im Bild 5.48 funktioniert die Zusammenarbeit des *ZX Spectrum* mit dem Kassettenrecorder reibungslos und ohne eventuelle Rückkopplungen auch bei höheren Übertragungsraten. Es entfällt damit auch das lästige Umstecken von Kopfhörer- und Diodenstecker, wie es zur Vermeidung von Rückkopplungen etwa bei Verwendung des *Geracord* erforderlich war. Daneben bietet das Interface einen gepufferten Video/Audio-Ausgang für den Anschluß eines Monitors.

Zur Funktion der Schaltung: Beim Abspeichern von Daten gibt die CPU die Daten seriell über die Datenleitung D3 an den ULA-Chip. Wird ein Flipflop (IC4) an diese Datenleitung geschaltet und das Einspeichern über IC3 von $\overline{\text{IORQGE}}$ und $\overline{\text{WR}}$ abhängig gemacht, steht am Ausgang des Flipflops genau das Signal, das auch an der MIC-Buchse ansteht, allerdings mit steileren Flanken und (über R16) einstellbarem Pegel. Beim Einlesen wird das Signal vom Kassettenrecorder durch den Operationsverstärker IC1 in ein Rechtecksignal umgewandelt und über den Tri-State-Treiber IC2 genau dann auf die Datenleitung D6 geschaltet, wenn die CPU die Daten vom ULA-Chip lesen will.

Dieser Zustand ist durch die aktivierten Signale $\overline{\text{IORQGE}}$ und $\overline{\text{RD}}$ gekennzeichnet, die ebenfalls von IC3 ausgewertet werden. Der Audio-Anschluß für den Monitor arbeitet wie der Save-Kanal, allerdings unter Verwendung der Datenleitung D4. Der Video-Anschluß wird über VT1 gepuffert, wie es bereits im Abschnitt 5.3.

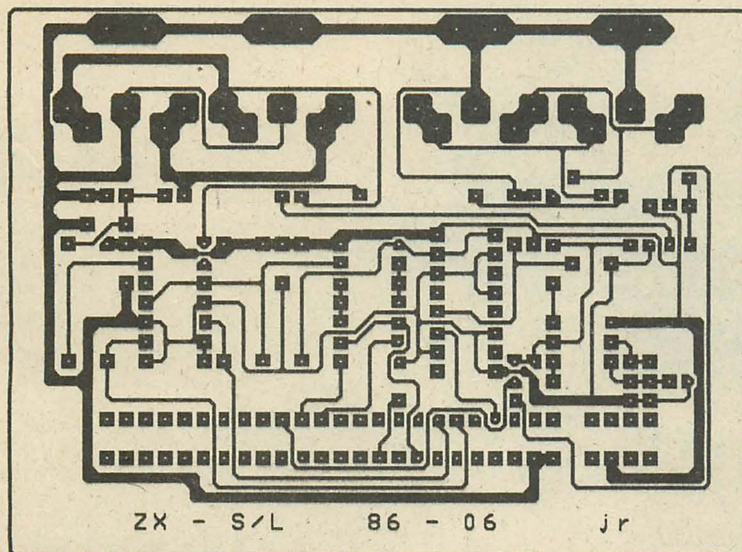


Bild 5.49 Layout für das SAVE/LOAD-Moduls

beschrieben wurde. Der Abgleich der Schaltung beschränkt sich auf die Einstellung von R16 für den Aufnahmepegel und R15 für den Audio-Pegel.

Zur Verringerung der Flankensteilheit der Impulse am Ausgang Audio Out kann erforderlichenfalls C3 etwas vergrößert werden. Zu steile Flanken führen insbesondere bei Recordern mit Aussteuerungsautomatik zu einem "Zustopfen" des Eingangsverstärkers. Die Interface-Platine nach Bild 5.49 ermöglicht die Parallelschaltung von jeweils zwei

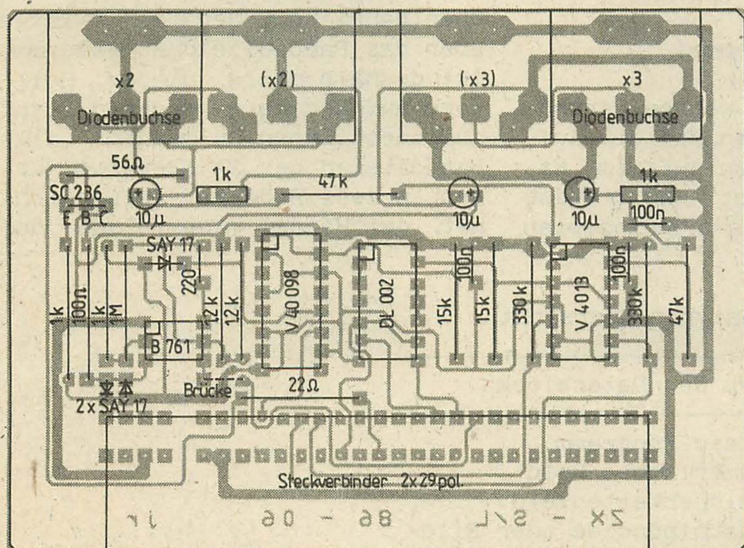


Bild 5.50 Bestückungsplan

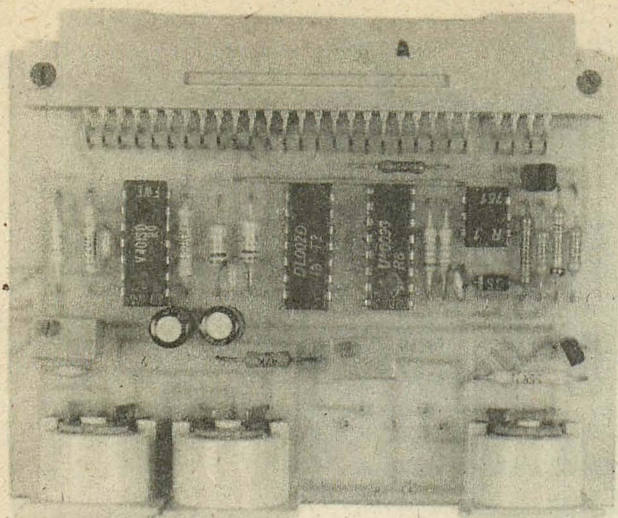


Bild 5.51 Musteraufbau

Buchsen für Tonband (X3) bzw. Monitor (X2). Sofern das nicht gewünscht wird, braucht nur die jeweils erforderliche Buchse bestückt zu werden. Spezielle Software ist nicht erforderlich, das Interface arbeitet mit den normalen Tape-Befehlen Save, Load, Merge usw.

5.8.1. Das Bandformat

Es gibt zwar nur wenige grundsätzliche Verfahren für die Aufzeichnung von Daten auf dem Magnetband, die Art und Weise aber, wie diese Grundprinzipien verwendet werden, ist von Computer zu Computer unterschiedlich. Infolgedessen besteht zwischen verschiedenen Kleincomputertypen nur selten Kompatibilität der Bandaufzeichnungen. Insbesondere für den Daten- bzw. Programmaustausch über größere Entfernungen ist aber die Aufzeichnung auf einer Magnetbandkassette nahezu unentbehrlich. Genaues Wissen über das Bandaufzeichnungsformat ist deshalb nötig, um die Software auf Rechnern anderen Typs zumindest lesen zu können. Alle Banddateien des *ZX Spectrum* werden in zwei Blöcken aufgezeichnet, dem Header (Kopf) und dem

Tabelle 5.12 Datentypen-Übersicht

Codierung	Typ des Datenblocks
0	Basic-Programm
1	Numerisches Feld
2	Zeichenkettenfeld
3	Maschinencode oder Bildschirminhalt

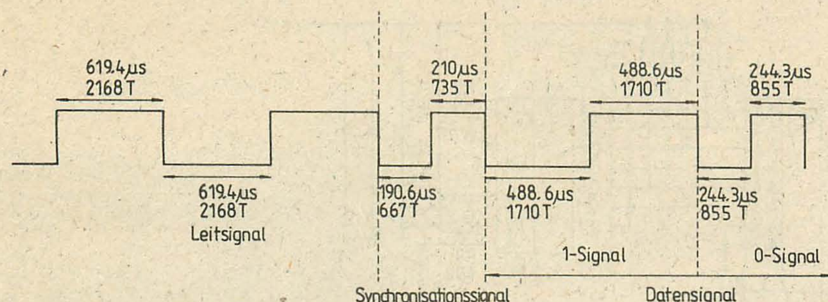


Bild 5.52 Signalfolge der Bandaufzeichnung des ZX Spectrum

Tabelle 5.13 Verwendung der Bytes 14 bis 17 im Header

Typ	Byte 14	Byte 15	Byte 16	Byte 17
0	Autostart-Zeilennummer		Nummern der Bytes	
1/2	-------	Feldname	-------	-------
3	Datenanfangsadresse		-------	-------

Datenblock. Der Header ist aus aus 19 Bytes aufgebaut. Das erste Byte des Headers ist stets Null (Kennbyte). Das zweite Byte enthält Informationen über den Typ des auf den Header folgenden Datenblocks entsprechend der Tabelle 5.12.

Die nächsten 10 Bytes enthalten den Dateinamen als ASCII-Zeichen, es folgen 2 Bytes zur Speicherung der Länge des folgenden Datenblocks. Die Bytes 14 bis 17 werden je nach Datenblocktyp unterschiedlich verwendet, wie der Tabelle 5.13 entnommen werden kann.

Das letzte Byte, das sogenannte Parity-Byte, wird zu Prüfzwecken beim Laden und Speichern der Banddateien verwendet.

Der Datenblock ist einfacher als der Header aufgebaut. Er enthält zwei Bytes mehr, als in der Längeninformation des Headers angegeben: das führende Type-Byte und das Parity-Byte am Ende. Nach dem Type-Byte folgen die Datenbytes unmittelbar aufeinander. Jeder Block beginnt mit einem Leitton von etwa 800 Hz, dessen Ende durch einen Synchronisationspuls mit einer Low-Zeit von 190,6 µs und einer High-Zeit von 210 µs markiert wird. Für jedes Datenbit wird eine vollständige Rechteckschwingung abgelegt. Sie dauert 488,6 µs für ein Low-Bit bzw. 977,2 µs für ein High-Bit. Das Aufzeichnungsverfahren ist sehr zuverlässig, dafür aber nicht besonders schnell [6].

5.9. Anwendungen von Kleincomputern in der Meß- und Prüftechnik

5.9.1. 8-Kanal-Digitalvoltmeter

Die Schaltung entsprechend Bild 5.53 zeigt ein vollständiges Digitalvoltmeter für dem Meßbe-

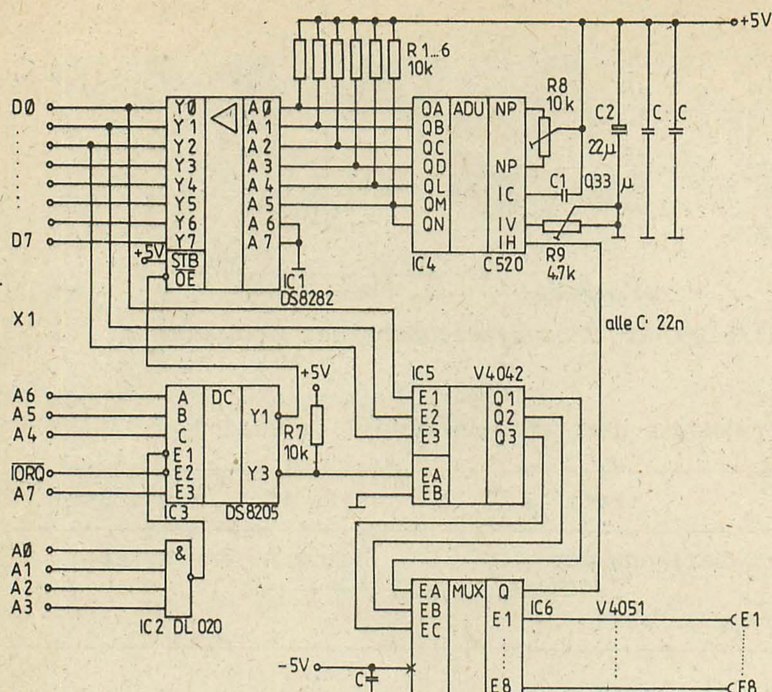


Bild 5. 53 Schaltung des Digitalvoltmeters

Meßbereich	R_A	R_B	R_C	R_D	R_E
bis 1 V	-	-	-	-	-
bis 10 V	-	-	9 MΩ	1 MΩ	-
bis 100 V	-	9 MΩ	900 kΩ	100 kΩ	-
bis 1000 V	9 MΩ	900 kΩ	90 kΩ	10 kΩ	-
bis 10 mA	-	-	-	-	100 Ω
bis 100 mA	-	-	-	1 Ω	9 Ω
bis 1000 mA	-	-	-	1 Ω	-

Der 1-Ω-Widerstand im Meßbereich 1000 mA muß mit mindestens 1 W belastbar sein!

reich von -99 mV bis +999 mV und einen elektronischen Meßstellenumschalter für 8 Eingänge zum Anschluß an den *ZX Spectrum*. Als AD-Wandler wird der *C 520* eingesetzt. Dieser älteste Vertreter der monolithisch integrierten AD-Wandler ist mittlerweile fast vollständig durch das System *C 500* abgelöst. Aufgrund seiner

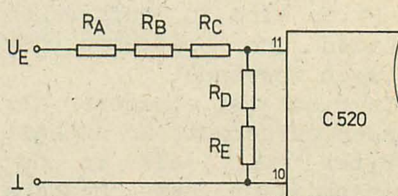


Bild 5. 54 Spannungsteilervorsatz für Vielfachmesser

Tabelle 5.14 Treiberprogramm für das Digitalvoltmeter (Basic-Teil)

```
 10 LET PU=64200
 20 DIM B$(8,17)
 30 FOR I=1 TO 8
 40   PRINT AT 20,0;"SPANNUNG ";I
 50   INPUT "BEMERKUNG:";B$(I)
 60 NEXT I
 70 PRINT AT 20,0;"              "
 80 FOR I=1 TO 8
 90   PRINT AT 2*I+2,0;"U(";I;") =      V  ";B$(I)
100 NEXT I
110 LET I=1
120 PRINT AT 2*I+2,2; INVERSE 1;I
130 PRINT AT 2,17;"BEMERKUNGEN"
140 PRINT AT 20,0;"KANALWECHSEL: ZIFFER 1 BIS 8"
150 PRINT AT 21,0;"JEDE ANDERE TASTE: ABBRUCH"
160 LET J=I
170 OUT 207,I-1
180 RANDOMIZE USR 64000
190 IF PEEK PU=58 THEN  POKE PU,45
      REM NEG.VZ
200 IF PEEK PU AND PEEK (PU+1) AND PEEK (PU+2)=59 THEN
      POKE PU,43
      POKE (PU+1),43
      POKE (PU+2),43
      REM POS.UEBERLAUF
210 IF PEEK (PU+1) AND PEEK (PU+2)=58 THEN
      POKE (PU+1),45
      POKE (PU+2),45
      REM NEG.UEBERLAUF
220 IF PEEK PU=48 THEN
      POKE PU,32
230 PRINT AT 2*I+2,7;
   CHR$ PEEK PU;CHR$ PEEK (PU+1);CHR$ PEEK (PU+2)
240 LET A$=INKEY$
250 IF A$=""THEN GO TO 160
260 IF VAL A$ 1THEN GO TO 240
270 IF VAL A$ 8THEN GO TO 240
280 LET I=VAL A$
290 PRINT AT 2*J+2,2;INVERSE 0;J
300 GO TO 120
310 CLEAR 63999
    LOAD "C520.COD"CODE
    CLS
    GO TO 0
320 REM SAVE AUF BAND
330 SAVE "C520.BAS"LINE 310
340 SAVE "C520.COD"CODE 64000,200
350 VERIFY "C520.BAS"
    VERIFY "C520.COD"CODE
```

Tabelle 5.15 Treiberprogramm für das Digitalvoltmeter (Maschinencode-Teil)

```
                ;******************
                ;*** DVM - CODE ***
                ;******************
                ;
FA00                        ORG  64000       ;ANFANGSADRESSE
FACA            PUFFER      EQU  64202       ;MESSWERTPUFFER
FA00 F5         ANFANG      PUSH AF
FA01 E5                     PUSH HL
FA02 21CAFA                 LD   HL,PUFFER   ;HL IST PUFFERZEIGER
FA05 DBEF       LSD         IN   A,(239)     ;MESSWERT EINLESEN
FA07 CB67                   BIT  4,A         ;LSD ?
FA09 20FA                   JR   NZ,LSD      ;NEIN: WARTEN AUF LSD
FA0B DBEF                   IN   A,(239)     ;JA: NOCHMAL EINLESEN
FA0D CB67                   BIT  4,A         ;UND ÜBERPRÜFEN
FA0F 20F4                   JR   NZ,LSD      ;WAR NUR STÖRNADEL
FA11 CD3BFA                 CALL MW          ;DIGIT ABSPEICHERN
FA14 DBEF       NSD         IN   A,(239)     ;MESSWERT EINLESEN
FA16 CB6F                   BIT  5,A         ;NSD ?
FA18 20FA                   JR   NZ,NSD      ;NEIN: WARTEN AUF NSD
FA1A DBEF                   IN   A,(239)     ;JA: NOCHMAL EINLESEN
FA1C CB6F                   BIT  5,A         ;UND ÜBERPRÜFEN
FA1E 20F4                   JR   NZ,NSD      ;WAR NUR STÖRNADEL
FA20 CD3BFA                 CALL MW          ;DIGIT ABSPEICHERN
FA23 DBEF       ENDNSD      IN   A,(239)     ;MESSWERT EINLESEN
FA25 CB6F                   BIT  5,A         ;NSD-IMPULS ZU ENDE ?
FA27 28FA                   JR   Z,ENDNSD    ;NEIN: SOLANGE WARTEN
FA29 DBEF       MSD         IN   A,(239)     ;MESSWERT EINLESEN
FA2B CB6F                   BIT  5,A         ;MSD ?
FA2D 20FA                   JR   NZ,MSD      ;NEIN: WARTEN AUF MSD
FA2F DBEF                   IN   A,(239)     ;NOCHMAL EINLESEn
FA31 CB6F                   BIT  5,A         ;UND ÜBERPRÜFEN
FA33 20F4                   JR   NZ,MSD      ;WAR NUR STÖRNADEL
FA35 CD3BFA                 CALL MW          ;DIGIT ABSPEICHERN
FA38 E1                     POP  HL
FA39 F1                     POP  AF
FA3A C9                     RET              ;RÜCKKEHR ZUM BASIC
FA3B CD47FA     MW          CALL ZS          ;ZEITSCHLEIFE
FA3E DBEF                   IN   A,(239)     ;MESSWERT EINLESEN
FA40 E60F                   AND  #0F         ;NUR UNTERE 4 BIT
FA42 F630                   OR   #30         ;30H ADDIEREN
FA44 77                     LD   (HL),A      ;ABSPEICHERN
FA45 2B                     DEC  HL          ;NÄCHSTER WERT
FA46 C9                     RET
FA47 3E0A       ZS          LD   A,10        ;A-REGISTER
FA49 3D         ZS1         DEC  A           ;DECREMENTIEREN
FA4A 20FD                   JR   NZ,ZS1      ;BIS A = 0
FA4C C9                     RET
FA4D                        END
```

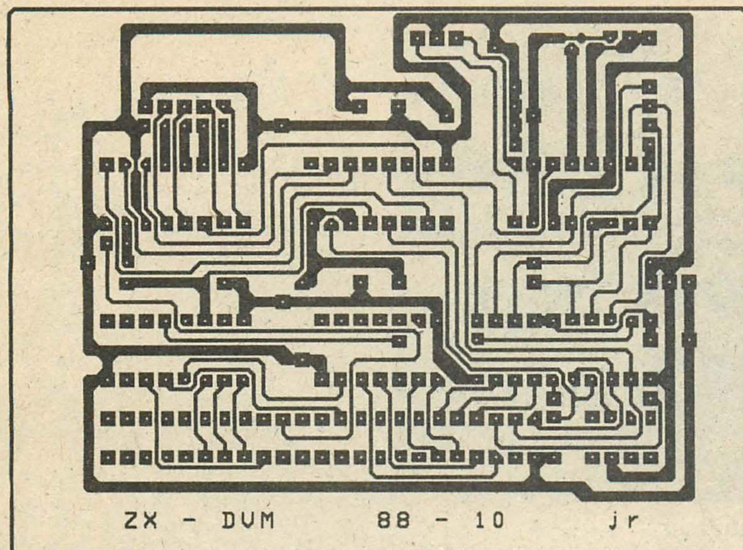


Bild 5.55 Layout für das Digitalvoltmeter

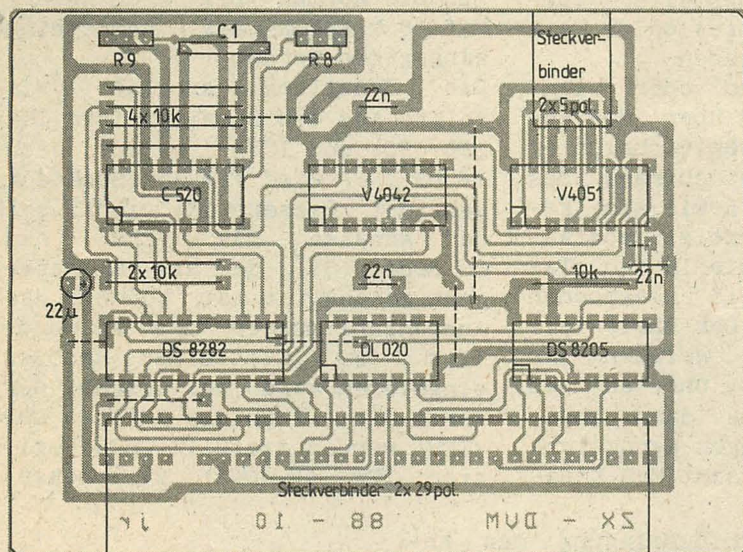


Bild 5.56 Bestückungsplan

guten Verfügbarkeit wird er hier dennoch verwendet.

Der *C 520* erlaubt in der angegebenen Beschaltung typisch etwa 100 Messungen pro Sekunde. Bei zyklischer Abfrage aller 8 Kanäle wird daher jeder Kanal unter Berücksichtigung der Programmlaufzeit nur etwa 10mal in jeder Sekunde abgefragt. Damit sind schnelle Spannungsänderungen nur bedingt erfaßbar. Sollen größere als die angegebenen Spannungen verarbeitet werden, sind ent-

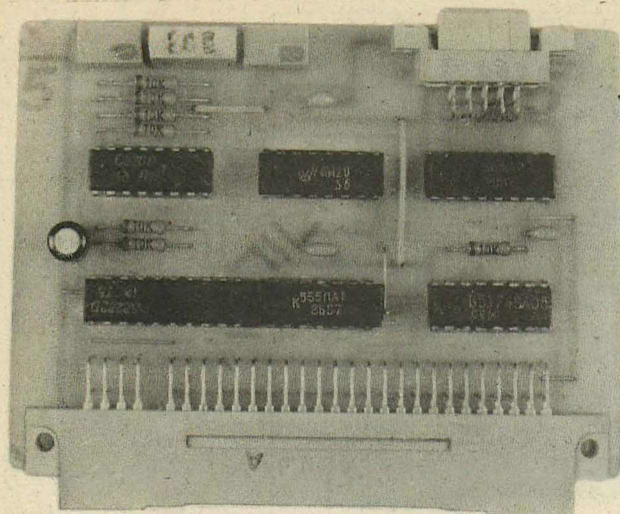


Bild 5.57 Musteraufbau

sprechende Spannungsteiler vorzusehen. Ein Beispiel dafür ist im Bild 5.54 angegeben.
Kleinere Spannungen oder Wechselspannungen sind über Meßverstärker bzw. Meßgleichrichter aufzubereiten. Die Abfrage des Digitalvoltmeters sowie die Kanalumschaltung erfolgt softwaregesteuert über die in den Tabellen 5.14 und 5.15 angegebenen Treiberprogramme. Der Basic-Teil wird mit LOAD "" geladen. Er startet sich selbst und lädt den CODE-Teil nach. Die dann abgeforderten Bemerkungen kennzeichnen lediglich die acht Meß-Kanäle. Sie können bei Nichtbedarf mit 8 x ENTER als Leerstring eingegeben werden.
Die Schaltung arbeitet wie folgt: Aus den Schaltkreisen IC2 (*DL 020*) und IC3 (*DS 8205* oder *74 LS 138*) wird ein Adreßdekoder gebildet. Dieser aktiviert bei der Adresse 239 (EFH) den *DS 8282* (IC1), der als Eingabeport geschaltet ist, wodurch das Ausgangsbitmuster des AD-Wandlers *C 520* (IC4) in den Rechner eingelesen wird. Bei Ausgabe der Werte 0 bis 7 an die Adresse 207 (CFH) werden diese im CMOS-Register IC5 (*V 4042*) zwischenge-

Tabelle 5.16 Anschlußbelegung des Anwendersteckverbinders des Digitalvoltmetermoduls

Kontakt	A-Ebene	B-Ebene
1	Eingang 5	Eingang 3
2	Eingang 7	Eingang 2
3	Eingang 8	Eingang 1
4	Eingang 6	Eingang 4
5	+5 V	Masse

speichert und liegen somit ständig am Meßstellenumschalter IC6 (*V 4051*) an. Damit ist die Durchschaltung von 8 verschiedenen Eingangsspannungen an den AD-Wandler möglich.
Der AD-Wandler *C 520* wird mit R8 bei kurzgeschlossenem Eingang und gestartetem Treiberprogramm auf Anzeige 000 mV und mit R9 bei angelegten z. B. 990 mV (mit Digitalvoltmeter kontrollieren) auf 990 mV Bildschirmanzeige abgeglichen. Dieser Abgleich ist wechselseitig zu wiederholen. Weitere Abgleicharbeiten sind nicht erforderlich.
Für die Einsteller R8 und R9 sollten Spindelpotentiometer und für den Kondensator C1 ein leckstromarmer Typ mit geringem TK (kein Elektrolytkondensator!) verwendet werden.

5.9.2. Schaltkreistester

Häufig stehen dem Elektronikamateur gebrauchte integrierte Schaltkreise zur Verfügung, über deren Funktionsfähigkeit Unklarheit besteht. In solchen Fällen wäre es wünschenswert, die Schaltkreise vor dem Einbau in die Schaltung zu überprüfen, da einerseits die nachträgliche Fehlersuche oft zeitraubend ist und andererseits das Auslöten defekter Schaltkreise die Platine beschädigen kann. Die Schaltung nach Bild 5.58 bietet eine Möglichkeit für die statische Überprüfung digitaler integrierter Schaltkreise. Kernstück der Schaltung ist eine PIO *U 855*, die über einen Adreßdekoder *U 74 HCT 138 DK* (anschlußkompatibel zu *DS 8205*) in Verbindung mit einem *DL 020 D* gemäß Tabel-

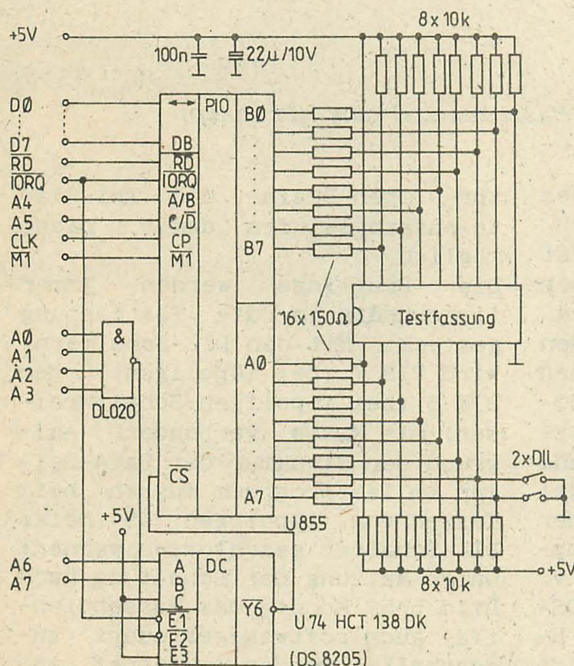


Bild 5.58 Schaltung des Schaltkreistesters

Tabelle 5.17 Adreßzuordnung des Schaltkreistesters

Port und Funktion	Adresse dez.	hex.
PIO Port A Daten	79	4F
PIO Port B Daten	95	5F
PIO Port A Steuerung	111	6F
PIO Port B Steuerung	127	7F

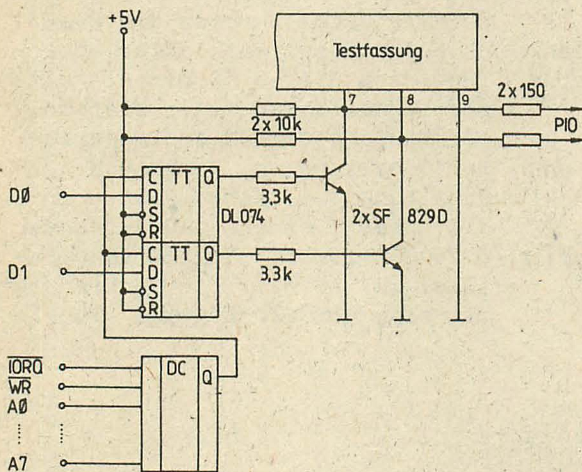


Bild 5.59 Softwaregesteuerte Massezuschaltung (Prinzip)

le 5.17 in den Adreßraum des *ZX Spectrum* eingeordnet ist.
An die Portleitungen der PIO ist eine 18polige Testfassung über 150-Ω-Widerstände angeschlossen. Diese Widerstände begrenzen den Strom in den Fällen, in denen eine als Ausgang definierte PIO-Portleitung mit einem Schaltkreisausgang verbunden ist und beide unterschiedliche Pegel führen. Sämtliche Anschlüsse der Testfassung liegen über Pull-up-Widerstände von 10 kΩ auf +5 V. Damit werden die für CMOS-Schaltkreise erforderlichen H-Pegel garantiert. Allerdings ist eine Erkennung von Open-Collector-, Open-Drain- und Tri-State-Ausgangsstufen damit nicht möglich.
Die Prüflinge werden immer linksbündig in die Testfassung gesteckt. Mit den DIL-Schaltern wird PIN 7 (bei 14poligen) oder PIN 8 (bei 16poligen Schaltkreisen) mit Masse verbunden. Aufgrund der Struktur der DATA-Zeilen im Testprogramm müssen beim Testen von 14poligen IS beide DIL-Schalter geschlossen werden!
Unter Nutzung der Schaltung nach Bild 5.59 könnte das Massepotential auch softwaregesteuert zugeschaltet werden. Hierbei arbeitet ein Flipflop *DL 074* als

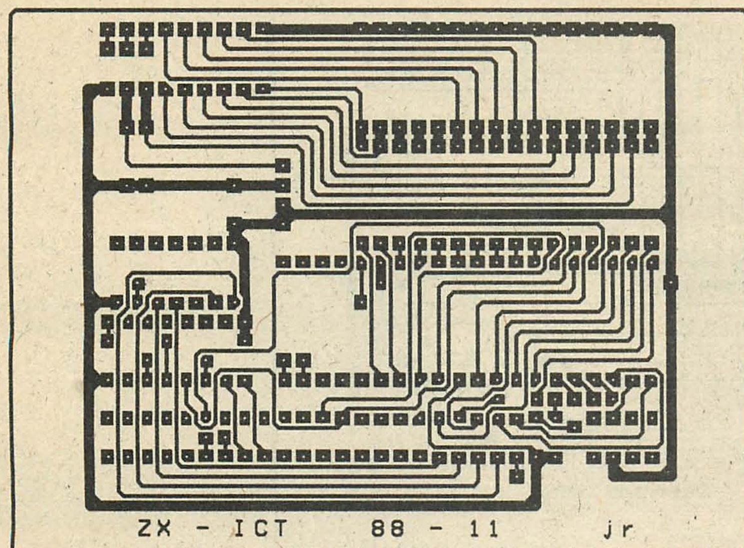


Bild 5.60 Layout (Leiterseite) für den Schaltkreistester

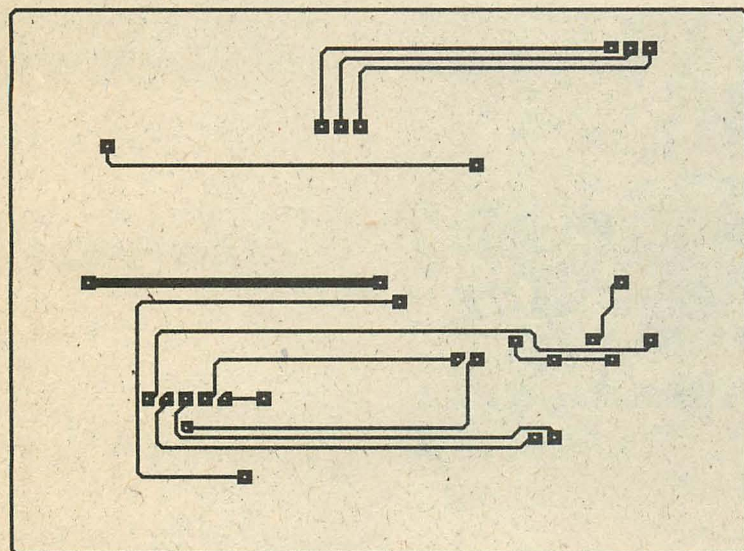


Bild 5.61 Layout (Bestückungsseite) für den Schaltkreistester

2-bit-Ausgabeport. Mit einer solchen Testanordnung und einem entsprechend modifizierten Treiberprogramm könnten dann beliebige 14- bis 18polige Schaltkreise selbständig erkannt und getestet werden. In Verbindung mit dem Prüfprogramm gemäß Tabelle 5.18 können folgende Schaltkreise sowie deren Äquivalenztypen getestet werden:
TTL-Typen: *7400, 7402, 7403, 7404, 7405, 7406, 7408, 7410, 7412, 7413, 7414, 7416, 7420,*

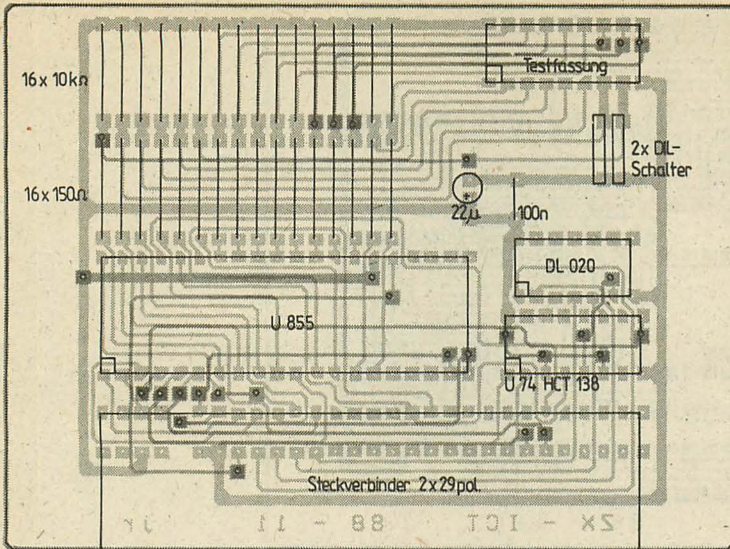


Bild 5.62 Bestückungsplan

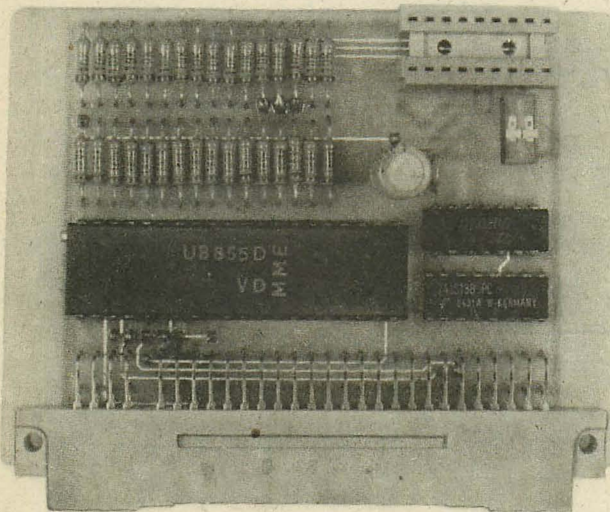


Bild 5.63 Musteraufbau

7421, *7426*, *7430*, *7433*, *7437*, *7438*, *7440*, *7446*, *7447*, *7450*, *7451*, *7453*, *7454*, *7474*, *7486*, *74128*, *74132*, *74136*, *74151*, *74153*, *74157*, *74192*, *74193*;
DS-Typen: *8205*;
CMOS-Typen: *4001*, *4011*, *4012*, *4028*.

Teilgeschädigte Schaltkreise werden als defekt klassifiziert. Zwar wäre mit geringem Aufwand eine (z. B. grafische) Ausgabe der defekten Gatter möglich, doch neigen teilgeschädigte IS eher zu Totalausfällen, so daß von ihrer Verwendung abgeraten wird. Die Schaltkreisbibliothek

Tabelle 5.18 Treiberprogramm für den Schaltkreistester

```
 10 REM IC-TESTER
 20 LET aco=111,bco=127,adat=79,bdat=95
 30 CSIZE 4,8
 40 LET af=0
    LET bf=0
    RESTORE
    CLS
    GO SUB 1100
 50 OUT aco,207
    OUT aco,255
    OUT bco,207
    OUT bco,255
 60 PRINT AT 17,21;"Typ bekannt ? (j/n)"
 70 LET t$=INKEY$
    IF t$=""THEN
      GO TO 70
 80 IF t$="j"OR t$="J"THEN
      GO TO 500
 90 IF t$="n"OR t$="N"THEN
      GO TO 110
100 GO TO 70
110 LET ai=IN adat
    LET bi=IN bdat
120 FOR l=1TO 5
130   READ av
      READ bv
140   IF (av  ai)OR (bv  bi)THEN GO TO 160
150   GO TO 180
160 NEXT l
170 IF l=6THEN
      GO TO 490
180 GO SUB ON l;190,200,210,220,230
    GO TO 310
190 RESTORE 250
    RETURN
200 RESTORE 260
    RETURN
210 RESTORE 270
    RETURN
220 RESTORE 280
    RETURN
230 RESTORE 300
    RETURN
240 DATA 27,219,31,187,31,251,63,251,21,171
250 DATA "7400",228,39,3,27,219,54,111,45,183
260 DATA "7410",224,71,4,31,187,47,95,54,239,57,247
270 DATA "7420",224,7,5,31,251,58,95,57,159,51,207,43,215
280 DATA "7430",192,7,10,63,251,63,39,63,71,31,103
290 DATA 47,103,55,103,59,103,61,103,62,103,63,99
300 DATA "7404",234,87,2,42,87,21,171
310 READ t$
```

```
320 PRINT AT 15,15;"Schaltkreis aequivalent ";t$
330 READ ap
    READ bp
340 OUT aco,207
    OUT aco,AND(ap,255)
    OUT bco,207
    OUT bco,AND(bp,255)
350 READ za
    FOR z=1TO za
360   READ ao
      READ bo
370   OUT adat,AND(ao,255)
      OUT bdat,AND(bo,255)
380   LET ai=IN adat,bi=IN bdat
390   LET af=OR(af,(XOR(ai,ao)))
400   LET bf=OR(bf,(XOR(bi,bo)))
410 NEXT z
420 IF af=0AND bf=0THEN
      PRINT AT 17,17;"IC ist funktionstuechtig"
      GO TO 440
430 PRINT AT 17,21;" IC ist defekt !     "
440 PRINT AT 20,20;"Neuer Test? (j/n)"
450 LET a$=INKEY$
460 IF a$="j"OR a$="J"THEN
      GO TO 40
470 IF a$="n"OR a$="N"THEN
      STOP
480 GO TO 450
490 CLS
    PRINT AT 15,15;"Schaltkreis nicht identifizierbar !"
    GO TO 1090
500 PRINT AT 17,21;"Typ eingeben:           ";
    INPUT t$
510 LET ty=VAL t$
520 IF ty=7450THEN
      RESTORE 530
      GO TO 310
530 DATA "7450",224,7,3,52,215,43,79,32,99
540 IF ty=7451THEN
      RESTORE 550
      GO TO 310
550 DATA "7451",224,7,3,31,251,52,151,43,15
560 IF ty=7453THEN
      RESTORE 570
      GO TO 310
570 DATA "7453",192,7,3,63,187,52,215,11,79
580 IF ty=7454THEN
      RESTORE 590
      GO TO 310
590 DATA "7454",192,7,3,31,155,20,151,11,15
600 IF ty=7400OR ty=7437OR ty=74132OR ty=7403THEN
      LET l=1
      GO TO 180
```

```
610 IF ty=7437OR ty=7438OR ty=7426THEN
     LET l=1
     GO TO 180
620 IF ty=7410OR ty=7412THEN
     LET l=2
     GO TO 180
630 IF ty=7420OR ty=7413OR ty=7440THEN
     LET l=3
     GO TO 180
640 IF ty=7430THEN LET l=4
     GO TO 180
650 IF ty=7404OR ty=7405OR ty=7406THEN
     LET l=1
     GO TO 180
660 IF ty=7414OR ty=7416THEN
     LET l=5
     GO TO 180
670 IF ty=7408THEN
     RESTORE 680
     GO TO 310
680 DATA "7408",228,39,3,63,255,18,75,9,147
690 IF ty=7446OR ty=7447THEN
     RESTORE 700
     GO TO 310
700 DATA "7447",136,255,13,24,1,4,255,28,65,92,231,13,137
710 DATA 93,131,30,39,94,19,31,17,95,199,60,1,124,3,61,185
720 IF ty=7474THEN
     RESTORE 730
     GO TO 310
730 DATA "7474",240,15,7,17,139,40,23,43,215,31,251,27,219
740 DATA 25,155,45,183
750 IF ty=74192THEN
     RESTORE 760
     GO TO 310
760 DATA "74192",230,49,7,119,183,16,89,24,57
770 DATA 16,25,92,57,76,41,16,25
780 IF ty=74193THEN
     RESTORE 790
     GO TO 310
790 DATA "74193",230,49,7,119,183,16,89,24,57,16,25,126
800 DATA 57,110,41,16,25
810 IF ty=8205THEN
     RESTORE 820
     GO TO 310
820 DATA "8205",192,255,11,96,127,97,191,98,223,99,239,100
830 DATA 247,101,251,102,253,39,255,111,255,119,255,71,255
840 IF ty=74153THEN
     RESTORE 850
     GO TO 310
850 DATA "74153",192,3,9,1,129,0,1,96,7,0,65,80,79,2,1,74
860 DATA 19,1,65,70,99
```

```
870 IF ty=7402OR ty=7433OR ty=74128THEN
      RESTORE 880
      GO TO 310
880 DATA "7402",201,147,4,9,147,36,39,18,75,54,79
890 IF ty=7486OR ty=74136THEN
      RESTORE 900
      GO TO 310
900 DATA "7486",228,39,4,0,3,54,183,45,111,27,219
910 IF ty=7421THEN
      RESTORE 920
      GO TO 310
920 DATA "7421",224,7,5,15,223,23,123,29,187,30,235,59,243
930 IF ty=74151THEN
      RESTORE 940
      GO TO 310
940 DATA "74151",176,1,9,24,1,20,9,18,5,17,13,16,131,16,75
950 DATA 16,39,16,31,104,1
960 IF ty=74157THEN
      RESTORE 970
      GO TO 310
970 DATA "74157",200,19,9,0,1,36,37,90,91,126,127,1,1,109
980 DATA 55,19,73,127,127,55,237
990 IF ty=4028THEN
      RESTORE 1000
      GO TO 310
1000 DATA "4028",255,195,11,4,1,0,69,2,33,0,165,1,17,32,21
1010 DATA 64,49,8,53,0,11,16,13,0,61
1020 IF ty=4001THEN
       RESTORE 1030
       GO TO 310
1030 DATA "4001",204,51,4,12,51,17,71,34,139,50,207
1040 IF ty=4011THEN
       RESTORE 1050
       GO TO 310
1050 DATA "4011",204,51,4,12,51,29,111,46,187,51,207
1060 IF ty=4012THEN
       RESTORE 1070
       GO TO 310
1070 DATA "4012",193,131,6,1,131,3,155,5,147,9,163,17,195,3
1080 CLS
     PRINT AT 15,8;"Dieser Typ ist nicht ";
     PRINT "im Testprogramm enthalten !"
1090 FOR i=0TO 500
     NEXT i
     GO TO 40
1100 CLS
     PRINT AT 0,0;CSIZE 12,12;OVER 1;" I C - T E S T E R "
1110 PRINT
     PRINT
1120 PRINT "Das Programm ermoeglicht die schnelle ";
     PRINT "Ueberpruefung der IC-Typen"
     PRINT
```

```
1130 PRINT "7400   7402   7403   7404   7405   7406   ";
     PRINT "7408   7410   7412"
1140 PRINT "7413   7414   7416   7420   7421   7426   ";
     PRINT "7430   7433   7437"
1150 PRINT "7438   7440   7446   7447   7450   7451   ";
     PRINT "7453   7454   7474"
1160 PRINT "7486   74128  74132  74136  74151  74153  ";
     PRINT "74157  74192  74193"
1170 PRINT
     PRINT "DS-Typen :    8205"
1180 PRINT
     PRINT "CMOS-Typen :  4001   4011   4012   4028   "
1190 PRINT
     PRINT "Die Typen 7400,7404,7410,7420,7430 bzw. ";
     PRINT "die Aequivalente werden"
1200 PRINT "erkannt. Fuer andere IC's ist der ";
     PRINT "Schaltkreistyp einzugeben"
1210 PRINT
     PRINT "Es koennen alle TTL- und CMOS-Reihen ";
     PRINT "getestet werden."
1220 PRINT "Durch Einfuegen weiterer DATA-Anweisungen ";
     PRINT "koennen neue"
1230 PRINT "IC-Typen in das Programm aufgenommen werden."
1240 PRINT
1250 PRINT "Testbeginn    ";CSIZE 12,12;"T"
1260 LET a$=INKEY$
     IF a$="t"OR a$="T"THEN
       GO TO 1280
1270 GO TO 1260
1280 CLS
     RETURN
```

Tabelle 5.19 Aufbau einer DATA-Zeile für den IC-Tester

Funktion			Beispiel
DATA	Schlüsselwort		DATA
ty	Typ-String		"7400"
ap	E/A-Maske PIO-Port A		228
bp	E/A-Maske PIO-Port B		39
za	Anzahl der Testschritte		3
ao	Testbitmuster Port A		27
bo	Testbitmuster Port B		219
ao	Testbitmuster Port A		54
bo	Testbitmuster Port B		111
.	.	za	45
.	.	ma	183
.	.		
ao	Testbitmuster Port A		
bo	Testbitmuster Port B		

Tabelle 5.20 Aufbau der Bitmuster für 14polige Schaltkreise

E/A-Maske Port A:	1 1 X X X X X X
E/A-Maske Port B:	X X X X X X 1 1
Testbitmuster Port A:	0 0 X X X X X X
Testbitmuster Port B:	X X X X X X 1 1

X = 0 für IS-Eingang, X = 1 für Ausgang

Tabelle 5.21 Aufbau der Bitmuster für 16polige Schaltkreise

E/A-Maske Port A:	1 X X X X X X X
E/A-Maske Port B:	X X X X X X X 1
Testbitmuster Port A:	0 X X X X X X X
Testbitmuster Port B:	X X X X X X X 1

X = 0 für IS-Eingang, X = 1 für Ausgang

Tabelle 5.22 Aufbau der Bitmuster für 18polige Schaltkreise

E/A-Maske Port A:	X X X X X X X X
E/A-Maske Port B:	X X X X X X X X
Testbitmuster Port A:	X X X X X X X X
Testbitmuster Port B:	X X X X X X X X

X = 0 für IS-Eingang, X = 1 für Ausgang

des Testprogramms läßt sich durch Einfügen weiterer DATA-Zeilen erweitern, deren Struktur den Tabellen 5.19 bis 5.22 zu entnehmen ist.
Die eigentliche Testroutine (Programmzeilen 370 bis 430) basiert auf der Tatsache, daß sich bei Abarbeitung eines IN-Befehls mit einem in Mode 3 (Bitbetrieb) arbeitenden PIO-Kanal die der CPU zugeführten Daten zusammensetzen aus den Daten des Eingaberegisters (das gilt für die Bits, die im E/A-Steuerwort als Eingänge definiert sind) und den Daten des Ausgaberegisters (für die Bits, die im E/A-Steuerwort als Ausgänge festgelegt sind).
Das Programm ist auf dem *ZX* Spectrum nur unter *Beta Basic* 3.0 lauffähig, weil dort die erforderlichen logischen Verknüpfungen implementiert sind. Diese Verknüpfungen kann man für andere Rechnertypen auch als Unterprogramme in Basic oder Maschinensprache schreiben.

5.10. Zusammenschaltung mehrerer Zusatzmodule

5.10.1. Busverlängerung

Der gleichzeitige Betrieb mehrerer Zusatzmodule an einem Kleincomputer gestaltet sich durch das Vorhandensein von meist nur einem Steckplatz problematisch. Einen Ausweg bietet die Busverlängerung nach Bild 5.64, mit der 5 Steckplätze für Hardware-Erweiterungen zur Verfügung stehen. Es werden dafür Buchsen- und Steckerleisten mit 2 x 29 Polen nach TGL 29 331 verwendet. Mit einer solchen Buchsenleiste sollte auch der *ZX Spectrum* ausgerüstet werden, um Kontakt- und

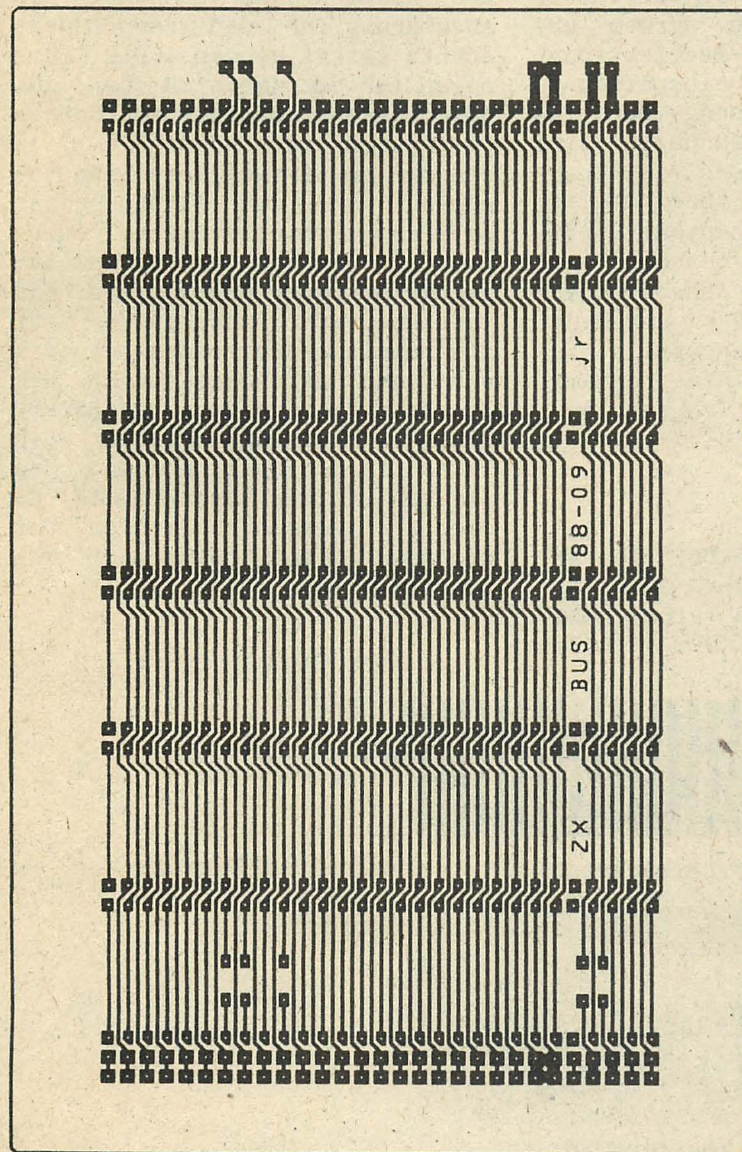


Bild 5.64 Layout für die Busverlängerung

Beschaffungsprobleme bei der Verwendung der Original-Steckverbinder zu umgehen.
Die Platine stellt durch die eng benachbarten, relativ langen Leitungen eine erhebliche kapazitive Belastung des Rechnerbusses dar, so daß man kaum ohne Bustreiber auskommen wird. Sämtliche mit der Spannungsversorgung des Computers in Verbindung stehenden Leitungen wurden unterbrochen, um die Betriebsspannungen auch extern einspeisen zu können. Bei Verwendung der intern erzeugten Spannungen sind diese Unterbrechungen zu überbrücken. Dann sind aber die im Abschnitt 5.1. genannten Stromergiebigkeiten zu beachten und Massekurzschlüsse, insbesondere mit -5 V und +12 V, unbedingt zu vermeiden. Anderenfalls besteht Gefahr für die dynamischen RAMs und den Transverter.

5.10.2. Adapter

Der Adapter (Bild 5.65) gestattet das Betreiben von Original-*Spectrum*-Zubehör an einem EGS-Steckverbinder mit 2 x 29 Polen in den Fällen, in denen eine Umrüstung des Interfaces nicht möglich ist. Hier sei nur die Ausleihe selten gebrauchter Zusatzgeräte (z. B. EPROM-Programmiergerät) genannt. Bild 5.65 zeigt nur eine Seite des Adapters. Die Platine ist auf der Leiter- und Bestückungsseite jeweils gleich strukturiert. In den Adapter muß zunächst die Aussparung für den Verpolungsschutz gesägt werden. Die Lötaugen für den Anschluß der 29. Anschlüsse sind nicht zu bohren oder gar zu verbinden. Sie bleiben für später geplante Erweiterungen oder den Aufbau einer Interrupt-Prioritätskette reserviert. Sehr zu empfehlen ist zwecks besserer Kontaktgabe das Verzinnen der Anschlüsse. Auf die metrische Seite des Adapters wird nun das Gegenstück des *Spectrum*-Verbinders gelötet. Sinnvoll ist der Einbau der Buchse in den *Spectrum* und des Steckers in die Zusatzmodule, da dann die Ausgangsleitungen des Rechners nicht unisoliert zugänglich sind.

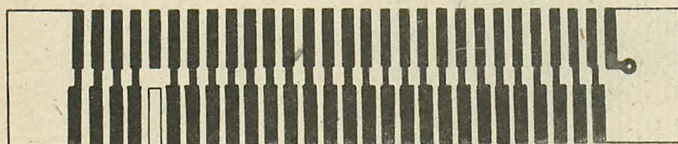


Bild 5.65 Adapter (Leiterseite)

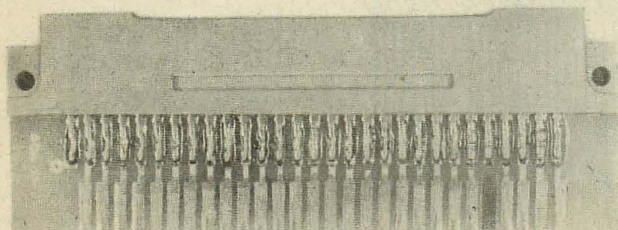


Bild 5.66 Adapter (Musteraufbau)

5.10.3. Bustreiber

Die Daten- und Adreßausgänge sowie die low-aktiven Steuerausgänge der CPU *U 880* sind in der Lage, je eine TTL-Lasteinheit zu treiben. Im Hinblick auf eine Vergrößerung der Busbelastbarkeit sowie auf die Unterdrückung von Störimpulsen und parasitären Kapazitäten können die CPU-Signale verstärkt werden. Das ist besonders dann erforderlich, wenn das System eine gewisse Größenordnung erreicht hat. Ab wann genau eine Buspufferung benötigt wird, hängt von vielen Faktoren ab und ist nicht zuletzt exemplarabhängig von Rechner zu Rechner verschieden.
Natürlich birgt der Einsatz eines Bustreibers auch Probleme. So muß man unter anderem die oft hohe Stromaufnahme beachten, die besonders beim Einsatz von

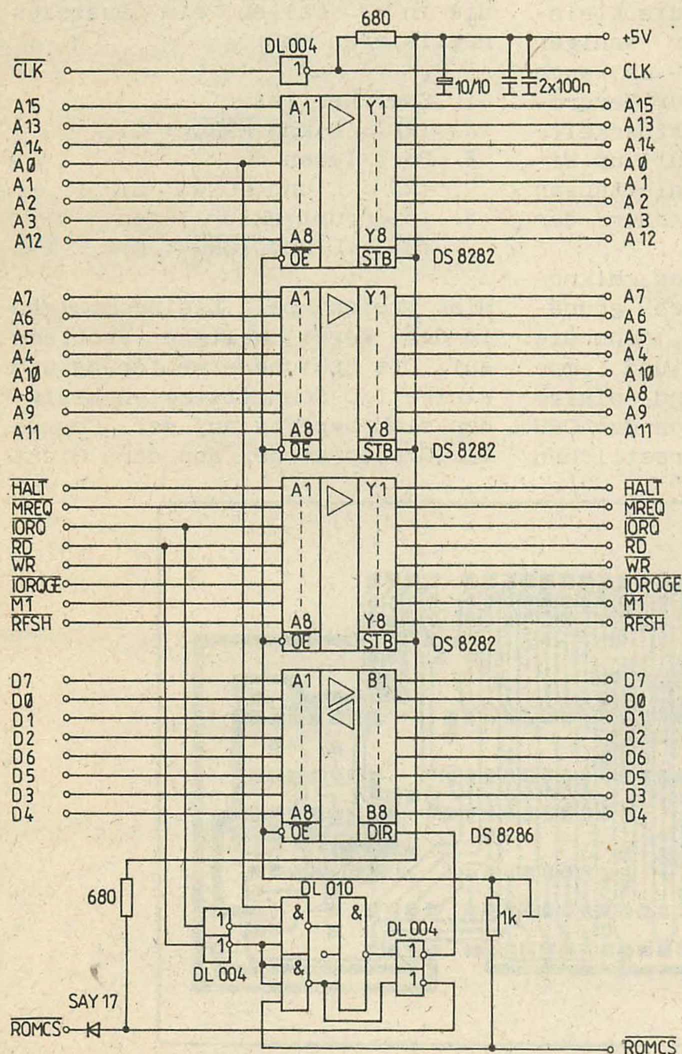


Bild 5.67 Schaltung des Bustreibers

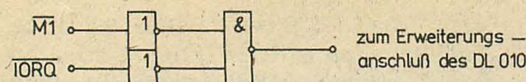


Bild 5.68 Erweiterungsschaltung zur Absicherung des Vektor-Interrupts

Schottky-Treibern (*DS*-Serie) schnell einige 100 mA erreichen kann, wofür meist schon ein externes Netzteil erforderlich wird. Bustreiber sind also nicht schlechthin das Mittel der Wahl, zumindest bezogen auf die Klein- und Heimcomputer. Für weniger hartnäckige Fälle gibt es auch einfachere Varianten zur Vergrößerung der Busbelastbarkeit, z. B. den Einbau von Pull-up-Widerständen in die Datenleitungen (8 x 4,7 bis 10 kΩ) oder der Einsatz einer CMOS-CPU.
Der Datenverkehr eines Mikrorechners vollzieht sich grundsätzlich bidirektional, denn die CPU kann Daten senden und empfangen. Die Adressen und einige Steuersignale werden von der CPU nur gesendet, die restlichen Steuersignale nur empfangen. Für die Datenleitungen ist deshalb ein in der Datenrichtung umschaltbarer Bustreiber erforderlich. Zur Datenrichtungsumschaltung dient eine Logikschaltung, die in 3 Fällen ein Buslesen realisieren muß:

1. Speicher lesen
   ($\overline{RD}$ = L und $\overline{MREQ}$ = L)
2. Port lesen
   ($\overline{RD}$ = L und $\overline{IORQ}$ = L)
3. Interruptvektor lesen
   ($\overline{M1}$ = L und $\overline{IORQ}$ = L)

Hier treten bei Kleincomputern in der Regel weitere Probleme auf. Die Speichererweiterung und einige I/O-Schnittstellen befinden sich bereits auf der Computer-Grundplatine, aus der Sicht

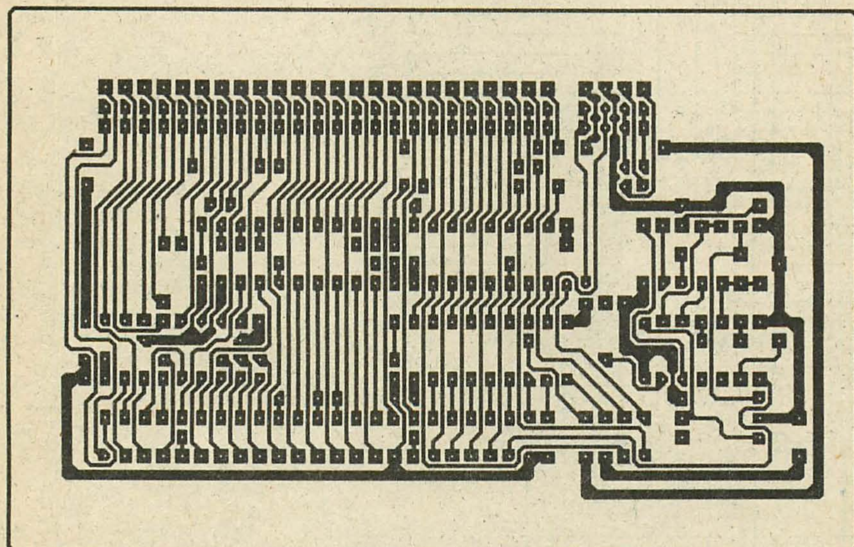


Bild 5.69 Bustreiber-Layout (Leiterseite)

der CPU also vor dem Bustreiber. Deshalb kommt es in den Betriebsarten Speicher lesen und Port lesen zu Buskonflikten, wenn die internen Systemeinheiten angesprochen werden und gleichzeitig der Bustreiber in Richtung CPU umgeschaltet wird. Es gibt mehrere Möglichkeiten, das zu umgehen:

1. Der Bustreiber wird von einem auf jedem Interface zu erzeugenden Richtungssignal gesteuert. Das vereinfacht die Hardware des Bustreibers zwar sehr, erfordert aber Eingriffe in jedes anzuschließende Interface, was besonders bei Industriegeräten mit unbekannter Schaltung problematisch ist.
2. Der Bustreiber wird ständig in Ausgaberichtung betrieben und nur bei Speicher- oder Portleseoperationen, die sich auf externe Systemeinheiten beziehen, in Eingaberichtung geschaltet.
3. Die Ausgänge des Bustreibers werden beim Lesen der internen Speicher bzw. I/O-Ports in den hochohmigen Zustand gebracht. Dazu dient der low-aktive Steuereingang $\overline{OE}$.

Die Variante 3 erwies sich besonders bei hohen Taktfrequenzen als etwas zeitkritisch. Darüber hinaus ist damit ein Betreiben des Rechners in der Interruptbetriebsart 2 nicht möglich. Dazu müßten die Ausgänge des Bustreibers ständig freigegeben sein (mit $\overline{OE}$ = L), um die Erkennung des Maschinenbefehls RETI durch die Peripheriebausteine zu gewährleisten. Deshalb wurde der Variante 2 der Vorzug gegeben. Unter Verwendung eines *DS 8286* und von 3 Schaltkreisen *DS 8282* wurde für den *ZX Spectrum* ein Bustreiber aufgebaut, dessen Schaltung das Bild 5.67 zeigt. Die Datenrichtung wird mit einer aus einem *DL 004* und einem *DL 010* bestehenden Schaltung umgeschaltet. Für die Taktpufferung wurde ein freies Gatter des *DL 004* verwendet, da der *ZX*

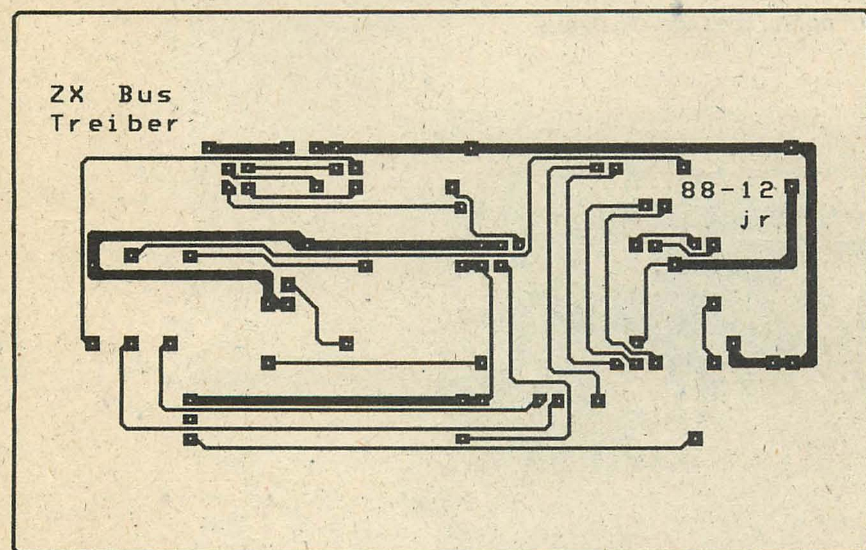


Bild 5.70 Bustreiber-Layout (Bestückungsseite)

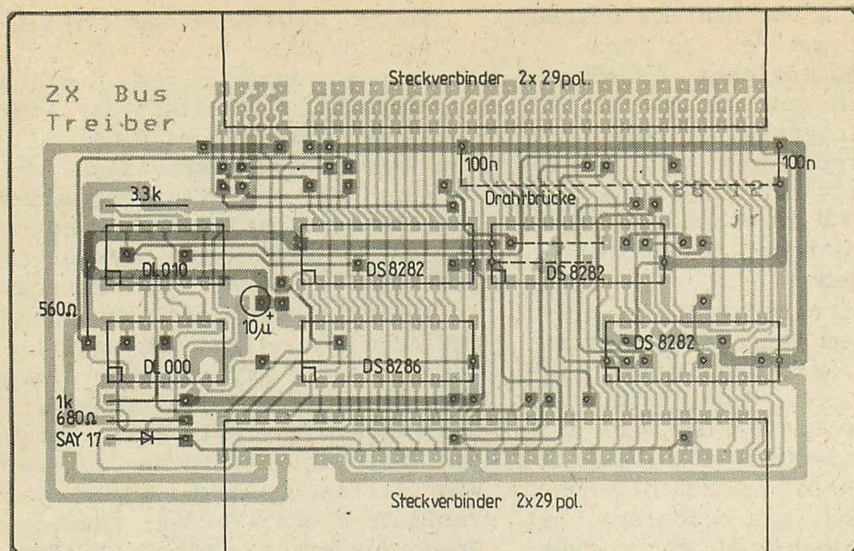


Bild 5.71 Bestückungsplan

Spectrum an seinem Bussteckverbinder ein invertiertes Taktsignal bereitstellt. Der 680-Ω-Widerstand sichert den erforderlichen Pegel für die Peripherieschaltkreise, deren Takteingänge abweichend von der TTL-Norm einen H-Pegel von +5 V benötigen. An den noch freien Eingang des *DL-010*-Gatters können bei Bedarf weitere Signalquellen zur Richtungsumschaltung des Bustreibers angeschlossen werden.
Im Bild 5.68 ist ein Beispiel zur Absicherung des Vektor-Interrupts (Interrupt-Mode 2) angegeben.

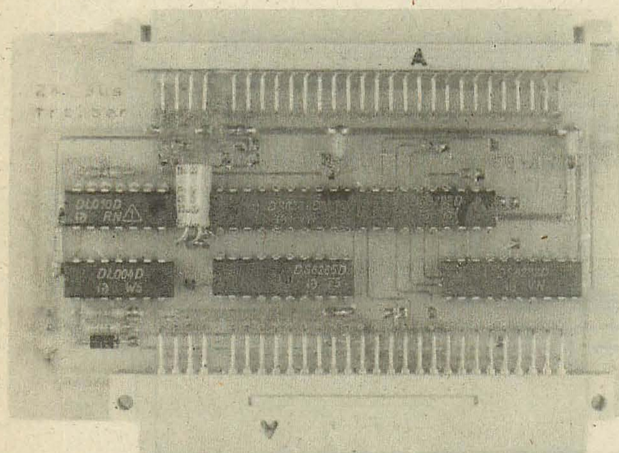


Bild 5.72 Musteraufbau

5.11. Speichererweiterungen

Die in den folgenden Abschnitten gezeigten Speichererweiterungen bedingen im Gegensatz zu allen anderen bisher vorgestellten Baugruppen Eingriffe in die bestehende Rechnerhardware. Zudem beziehen sie sich sehr stark auf die Originalversion des *ZX Spectrum* und sind daher nicht ohne größere Änderungen auf andere Rechnertypen übertragbar, weshalb sie bewußt am Schluß behandelt werden.

5.11.1. RAM-Erweiterung von 16 k auf 48 k

In der Grundversion des *ZX Spectrum* sind neben einem 16K-ROM, der Betriebssystem und BASIC-Interpreter enthält, 16 kByte RAM-Kapazität vorhanden. Hierin drängeln sich die Systemvariablen, der Druckerpuffer, der Stack und die Displaydatei, so daß für Anwenderprogramme nicht mehr viel .Platz bleibt. Auch wenn diese 16 k manchem Umsteiger vom *ZX 81* gigantisch anmuten (dort war für all das lediglich 1 kByte vorhanden), aus heutiger Sicht sind 16 k RAM eher als bescheiden anzusehen. Es liegt also nahe, den RAM-Bereich zu vergrößern, zumal der *ZX Spectrum* dafür schon vorbereitet ist. Auf der Rechner-Platine sind 12 Steckfassungen angeordnet, die nur mit den richtigen Schaltkreisen bestückt werden müssen. Es werden benötigt:

8 64K x 1 dynamische RAM *4164* (*U 2164*) als IC 15...22
1 Schaltkreis *74 LS 32* (*DL 032*) als IC 23
1 Schaltkreis *74 LS 00* (*DL 000*) als IC 24
2 Schaltkreise *74 LS 157* (*DL 257*) als IC 25, 26

In Klammern stehen die entsprechend verwendbaren DDR-Typen. Der Ersatz des *74 LS 157* durch den *DL 257* wurde an einigen Exemplaren des Spectrum praktisch überprüft. Trotzdem kann es in Einzelfällen zu Unverträglichkeiten kommen. Schaltkreise der Standard-TTL-Serie (wie *D 100*) sollten wegen der hohen Stromaufnahme nicht verwendet werden. Verwendbar sind neben *LS*-Typen auch Schaltkreise der *ALS*-, *HC*- und *HCT*-Serie. Die IC-Bezeichnungen findet man im Bestückungsdruck der Platine.

5.11.2. RAM-Erweiterung von 48 k auf 80 k

Im vorigen Abschnitt wurde gezeigt, wie mit geringem Aufwand der RAM-Rereich des *ZX Spectrum* um 32 kByte vergrößert werden kann. Dafür wurden als RAM-Schaltkreise 64K-Typen empfohlen, die dabei aber nur zur Hälfte ausgenutzt werden. Eine kleine Zusatzschaltung gestattet es jedoch, auch die anderen 32 kByte zu nutzen. Der *Spectrum* ist aber nunmehr mit einem 16K-ROM, 16K RAM und der 32K-RAM-Erweiterung aus dem Abschnitt 5.11.1. bestückt. Das bedeutet volle Belegung für den Speicher-Adreßraum des *U 880*, weil dieser Prozessor mit seinen 16 Adreßleitungen nur $2^{16} = 65536$ (= 64 k) Speicherplätze adressieren kann. Ein weiterer Ausbau des Speichers kann daher nur über eine Umschaltung von Speicherbereichen erfolgen. Die

Tabelle 5.23 Speicheraufteilung im ZX Spectrum

Typ	16-K-Version	48-K-Version	80-K-Version
ROM	0000...3FFF H	0000...3FFF H	0000...3FFF H
RAM	4000...7FFF H	4000...7FFF H	4000...7FFF H
RAM	-------------	8000...FFFF H	8000...FFFF H (BANK 1)
RAM	-------------	-------------	8000...FFFF H (BANK 2)

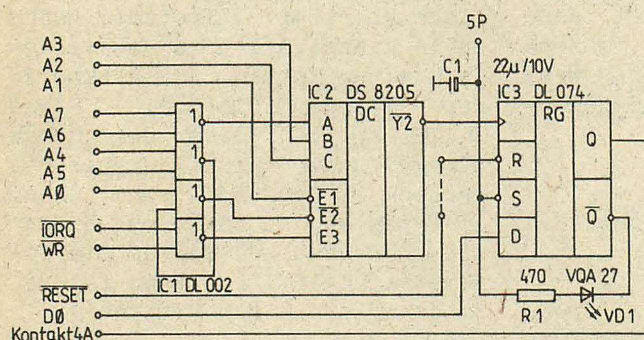


Bild 5.73 Elektronische Speicherbank-Umschaltung

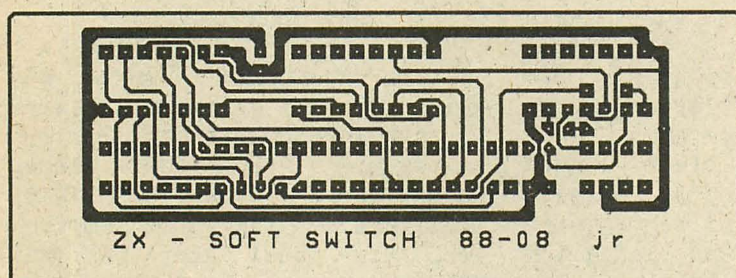


Bild 5.74 Layout für die Speicherbank-Umschaltung

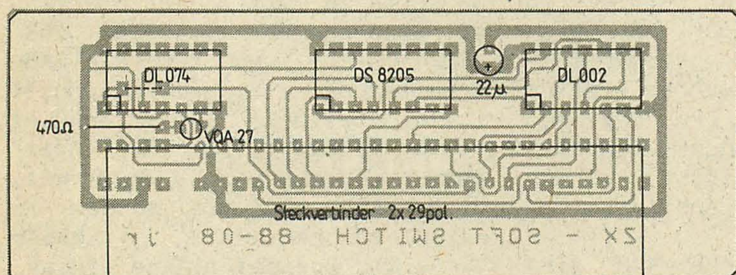


Bild 5.75 Bestückungsplan

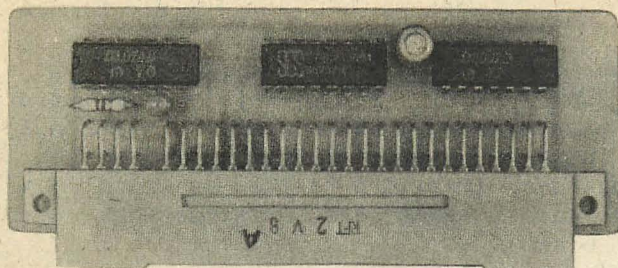


Bild 5.76 Musteraufbau

Tabelle 5.23 verdeutlicht die Speicheraufteilung der verschiedenen Versionen.
Die Umschaltung der Speicherbänke wird über die Schaltung nach Bild 5.73 elektronisch realisiert. Aus den Schaltkreisen *DL 002* und *DS 8205* (bzw. *74 LS 138* oder *74 HCT 138*) wird ein Adreßdekoder gebildet. Dieser aktiviert bei einem OUT-Befehl an die Adresse 0001 ein Flipflop (*DL 074*), das den aktuellen Zustand der Datenleitung D0 übernimmt. Der Ausgang dieses Flipflops gelangt über eine Leuchtdiode zur Anzeige und wird über eine bisher unbeschaltete Leitung am Bussteckverbinder in den *Spectrum* geführt. Verwendet wurde die Leitung B4. Diese Leitung ist auf der Platine des *ZX Spectrum* mit dem mittleren Lötauge rechts neben dem ULA-IC (siehe Bild 5.77) zu verbinden. Diese Angaben beziehen sich auf die relativ weit verbreitete Version 2 des *ZX Spectrum*. Bei anderen Versionen ist unter Zuhilfenahme der Schaltung des Rechners sinngemäß zu verfahren.
Unter dem *DL 074* sind auf der Modulplatine 2 Lötaugen angeordnet. Sie verbinden den RESET-Anschluß des *DL 074* mit dem System-RESET, falls hier eine Drahtbrücke eingelötet wird. Damit kann erreicht werden, daß nach einem Reset des Rechners immer die Speicherbank 0 eingeschaltet ist. Bei einem RESET durch Unterbrechen der Betriebsspannung ist das jedoch in jedem Fall gegeben.
Mit den Befehlen OUT 1,0 und OUT 1,1 kann zwischen den beiden Speicherbereichen umgeschaltet werden. Dabei muß RAMTOP allerdings unter den umschaltbaren Bereich gelegt werden. Ansonsten würde der Rechner beim Wechsel der Speicherbänke seinen Maschinenstapel "vergessen". Das führt zum Systemabsturz! Soll eine 48K-Version des *Spectrum* umgerüstet werden, so ist das immer dann unproblematisch, wenn für die Speicherschaltkreise Steckfassungen vorgesehen sind. Ein Auslöten der Schaltkreise setzt bei der filigranen Leiterplatte große Erfahrungen voraus und kann deshalb nicht generell empfohlen werden.

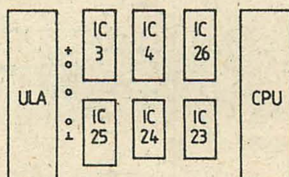


Bild 5.77 Ausschnitt aus der Platine des ZX Spectrum

5.11.3. ROM-Erweiterungen

Wie bereits dargelegt, kann die CPU *U 880* lediglich 64 K Speicher direkt adressieren. Deshalb ist auch dieser Abschnitt nicht im Sinne einer Vergrößerung des ROM-Bereiches zu verstehen. Auch hier geht es vielmehr um einen (eventuell softwaremäßig schaltbaren) Austausch des internen ROMs durch einen oder mehrere andere ROMs (oder EPROMs) gleicher oder geringerer Speicherkapazität. Diese ROMs könnten z. B. ein geändertes Betriebssystem oder eine andere Programmiersprache enthalten. Darüber hinaus ist es möglich, das interne ROM nur bei einem bestimmten Bitmuster auf dem Adreßbus (in Verbindung mit $\overline{MREQ}$ = 0) abzuschalten und dafür ein externes ROM einzublenden. Auf diese Weise können einzelne Routinen des Original-ROMs geändert werden, was für die Einbindung von Treibern für systemfremde Drucker oder Massenspeicher (Floppy-Disk) interessant werden kann. Darauf wird hier jedoch nicht näher eingegangen. Im Bild 5.78 ist die Anschlußbelegung des Original-ROMs des

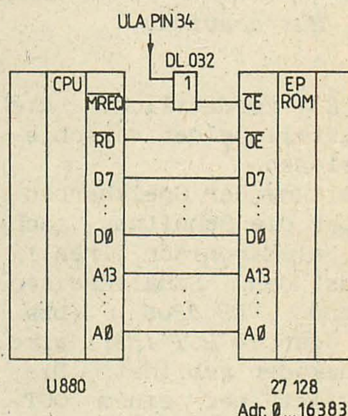


Bild 5.79 ROM-Ersatz durch 1 x 27128

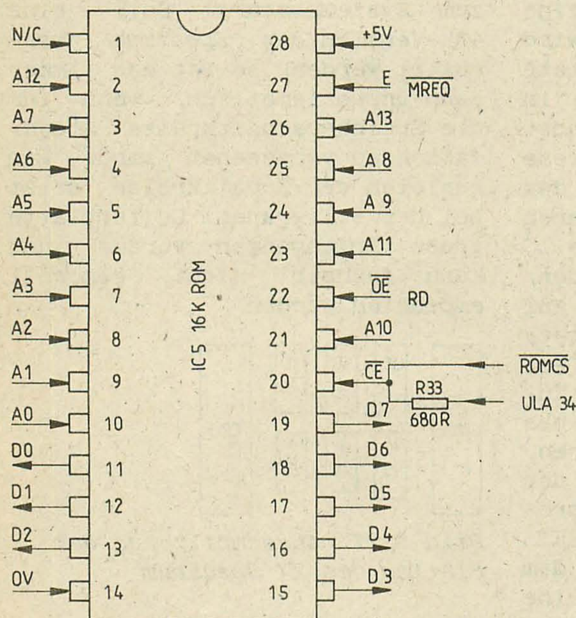


Bild 5.78 Anschlußbelegung des ZX Spectrum-ROM

Tabelle 5.24 Pinbelegung der wichtigsten EPROM-Typen

EPROM 16K x 8 27128	EPROM 8K x 8 2764	EPROM 4K x 8 2732	EPROM 2K x 8 2716	PIN 28	PIN 24	PIN 24	PIN 28	EPROM 2K x 8 2716	EPROM 4K x 8 2732	EPROM 8K x 8 2764	EPROM 16K x 8 27128
Upp	Upp			1			28			+5V	+5V
A12	A12			2			27			$\overline{PGM}$	$\overline{PGM}$
A7	A7	A7	A7	3	1	24	26	+5V	+5V	NC	A13
A6	A6	A6	A6	4	2	23	25	A8	A8	A8	A8
A5	A5	A5	A5	5	3	22	24	A9	A9	A9	A9
A4	A4	A4	A4	6	4	21	23	Upp	A11	A11	A11
A3	A3	A3	A3	7	5	20	22	$\overline{OE}$	$\overline{OE/Upp}$	$\overline{OE}$	$\overline{OE}$
A2	A2	A2	A2	8	6	19	21	A10	A10	A10	A10
A1	A1	A1	A1	9	7	18	20	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$
A0	A0	A0	A0	10	8	17	19	D7	D7	D7	D7
D0	D0	D0	D0	11	9	16	18	D6	D6	D6	D6
D1	D1	D1	D1	12	10	15	17	D5	D5	D5	D5
D2	D2	D2	D2	13	11	14	16	D4	D4	D4	D4
Masse	Masse	Masse	Masse	14	12	13	15	D3	D3	D3	D3

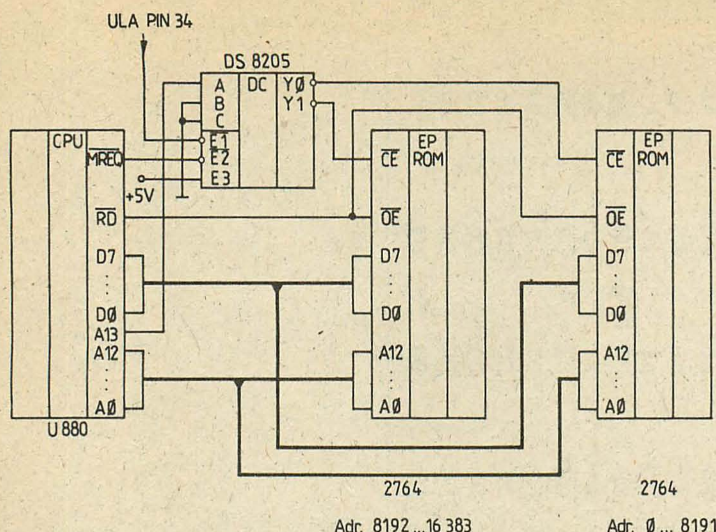


Bild 5.80 ROM-Ersatz durch 2 x 2764

ZX Spectrum. Die Anschlußbelegung der wichtigsten EPROM-Typen geht aus Tabelle 5.24 hervor. Man erkennt, daß das ROM bis auf die Anschlüsse 1 und 27 anschlußkompatibel zum EPROM *27128* ist. Pin 1 (beim ROM nicht beschaltet) ist lediglich für die Programmierung des EPROMs interessant und wird daher nicht näher betrachtet. Pin 27 des ROMs ist mit der CPU-Leitung MREQ verbunden, dem EPROM *27128* werden hier die Programmierimpulse zugeführt. Um dem angeschlossenen EPROM dennoch ein aktives MREQ-Signal mitzuteilen, wird es mit dem Signal CE (Pin 20) verknüpft. Das kann über ein Gatter eines *DL 032* erreicht werden. Sämtliche anderen Anschlüsse des EPROMs sind mit den entsprechenden Anschlüssen des ROMs zu verbinden. Da Beschaffung und Programmierung des EPROMs *27128* mitunter Schwierigkeiten bereiten, sind in den Bildern 5.79 bis 5.81 auch Schaltungslösungen unter Verwendung von EPROM-Typen kleinerer Speicherkapazität angegeben. Auf den Einsatz des 1k-EPROM *2708* (*U 555*) sollte schon wegen der unhandlichen Spannungsversorgung verzichtet werden, sofern der Rechner nicht über entsprechende Schaltungseinheiten verfügt.

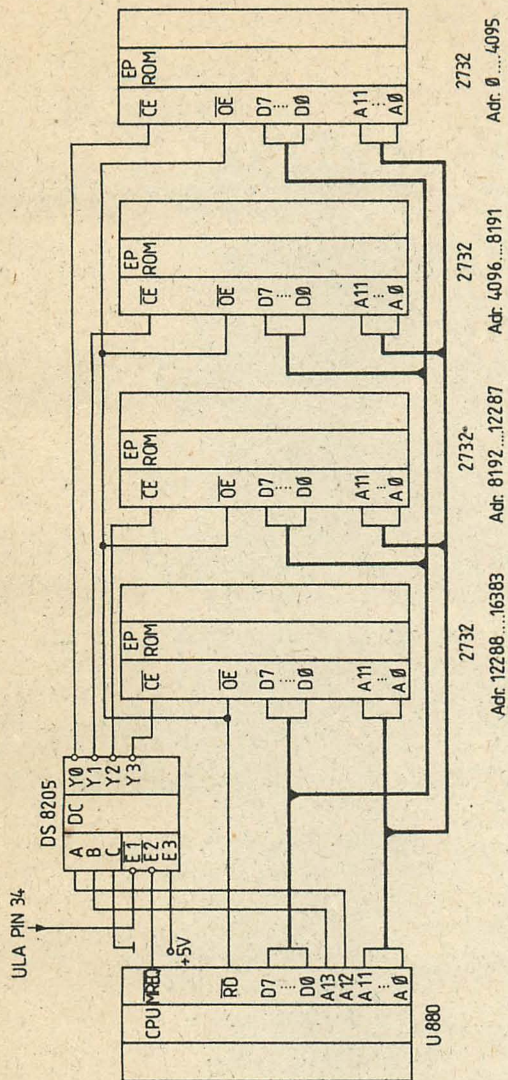


Bild 5.81 ROM-Ersatz durch 4 x 2732

6. Aufbauhinweise

Für den Aufbau der vorstehend beschriebenen Module und eine eventuelle Fehlersuche werden Grundkenntnisse der Mikroelektronik, insbesondere der Mikroprozessorfamilie *U 880* einschließlich des Signalspiels der CPU, sowie Erfahrungen beim Aufbau elektronischer Schaltungen und im Umgang mit elektronischen Bauelementen vorausgesetzt. Zur Inbetriebnahme der Platinen ist zumindest ein Vielfachmeßgerät, besser auch ein Oszilloskop, erforderlich. Die veröfffentlichten Leiterplatten weisen zumeist einen Schwierigkeitsgrad auf, der nur eine fotomechanische Herstellung sinnvoll erscheinen läßt. Dabei dürfte der Einzelamateur wahrscheinlich auf die Hilfe von Elektronik-Arbeitsgemeinschaften oder spezialisierten Firmen angewiesen sein.

Die Bohrungen sollten möglichst mit Spiralbohrern von 0,7 oder 0,8 mm Durchmesser mit einer Bohrmaschine hoher Drehzahl ausgeführt werden. Lediglich für Bauelemente mit dickeren Anschlüssen (Taster, Diodenbuchsen, Stellwiderstände usw.) sind entsprechend größere Bohrer (bis etwa 1,3 mm) zu wählen. Dem wurde bei der Platinengestaltung bereits mit größeren Lötaugen Rechnung getragen. Vor der Bestückung empfiehlt sich eine Reinigung der Platine mit einem Putzmittel und ein Einstreichen mit in Spiritus gelöstem Kolophonium. Ebenfalls vor der Bestückung sollten alle zum Steckverbinder führenden Leitungen mit einem Ohmmeter untereinander auf Feinschluß geprüft werden. Eventuell vorhandene Haarrisse in den Leiterzügen werden mit einer starken Lupe aufgespürt und durch Verzinnen beseitigt. Nach der Bestückung wird die Prüfung auf Feinschluß wiederholt, besonders bei den Daten und Adreßleitungen. Da die meisten Kleincomputer einen ungepufferten Bus aufweisen, kann es bei Nichtbeachtung dieser Vorsichtsmaßregeln zu Ausfällen durch unbeabsichtigtes Verbinden einzelner Signalleitungen kommen. Aus diesem Grund dürfen sämtliche Zusatzgeräte nur bei ausgeschalteter Betriebsspannung gesteckt bzw. abgezogen werden. Man vermeide auch grundsätzlich ein Zurechtrücken der Module bei eingeschaltetem Rechner! Nach vollständiger Bestückung ist die Stromaufnahme zu kontrollieren und die Betriebsspannung an allen Schaltkreisen nachzuweisen. Abgleicharbeiten an den Hardware-Zusätzen sind zumeist nicht erforderlich oder auf ein Minimum beschränkt.

7. Reparaturtips

Trotz aller Vorsicht kann es besonders den Hardware-Bastlern passieren, daß, vielleicht durch eine Unachtsamkeit, oft aber auch "einfach so", der Rechner streikt. Einige an der Originalversion des *ZX Spectrum* immer wieder auftretende Fehler und Möglichkeiten für deren Beseitigung werden im folgenden beschrieben.

Zur Eingrenzung des Fehlers werden zunächst die 3 Betriebsspannungen +5 V, -5 V und +12 V überprüft. Das geht sehr einfach an einem der 16K-RAMs. Hier liegen -5 V am Pin 1, +12 V am Pin 8, +5 V am Pin 9 und 0 V (Masse) am Pin 16. Fehlen alle Spannungen, wird der Fehler im Netzteil liegen (Vorsicht Netzspannung!). Fehlen +5 V, kann der Spannungsregler-Schaltkreis *7805* die Ursache sein. Dieser Typ liegt in der DDR nur als *Tesla*-Import in einem viel größeren Gehäuse vor. Es ist deshalb günstiger, für den *7805* einen Spannungsregler *B 3170* oder *B 3171* zu verwenden. Die dafür erforderlichen Widerstände zur Einstellung der Ausgangsspannung sind bequem noch im *Spectrum*-Gehäuse unterzubringen. Man beachte die unterschiedlichen Anschlußbelegungen der Schaltkreise!

Der mit Abstand häufigste Defekt ist der Ausfall des Transverters, der -5 V und +12 V für die 8 dynamischen 16K-RAMs erzeugt. Dieser Fehler tritt manchmal allein, oft aber in Verbindung mit dem Ausfall eines der 8 16K-RAMs auf. Zur Fehlersuche können die beiden Transistoren des Transverters ausgelötet und überprüft werden. In der Originalschaltung des *ZX Spectrum* werden die Typen *ZTX 651* (TR4 - npn) und *ZTX 213* (TR5 - pnp) verwendet. Im Fehlerfall können diese Transistoren durch die Typen *SF 816* bis *SF 819* bzw. *SF 826* bis *SF 829* ersetzt werden. Diese Nachfolgetypen der bekannten *SF 116* bis *SF 119* bzw. *SF 126* bis *SF 129* sind in einem potentialfreien Plastgehäuse verkappt, was bei dem gedrängten Aufbau vorteilhaft ist. Fehlen -5 V und +12 V nach Transistorwechsel noch immer, ist noch eines der 8 16K-RAMs defekt. Am einfachsten findet man den defekten Schaltkreis durch oszilloskopische Kontrolle der 8 Datenleitungen. Dabei ist es gelegentlich vorteilhaft, den RESET-Eingang der CPU über einen Widerstand von etwa 1 kΩ auf Masse zu legen. Die dem defekten RAM zugeordnete Datenleitung zeigt zeigt einen auffälligen Pegel.

Als Ersatztyp für die RAMs *4116* eignen sich *U 256* der DDR-Produktion. Sollte nach diesen Maßnahmen der Rechner noch nicht arbeiten, kann der Fehler noch an der CPU, dem ROM oder dem ULA liegen. Für den *Z 80* kann eine CPU *UA 880* eingesetzt werden. Oft arbeitet zunächst auch ein *UB*-Typ, der aber bei Erwärmung des Rechners aussetzen kann. Der ROM kann durch ein EPROM ent-

sprechenden Inhalts ersetzt werden. Ein defektes ULA läßt sich bei vertretbarem Aufwand nur durch einen Originalschaltkreis ersetzen.
Ein in Verbindung mit dem Interface I häufig auftretender Fehler ist das Fehlen des M1-Signals der CPU. Festgestellt wird der Fehler wieder durch oszilloskopische Kontrolle. Auch in diesem Fall ist ein CPU-Wechsel erforderlich.

8. Literaturverzeichnis

[1] VEB Mikroelektronik *Karl Marx* Erfurt, Mikroprozessorsystem der II. Leistungsklasse, Technische Beschreibung CPU, PIO, SIO, CTC und DMA, Erfurt 1984

[2] *H. Kieser/M. Meder*, Mikroprozessortechnik, Berlin 1986

[3] *H. Barthold/H. Bäurich*, Mikroprozessoren - Mikroelektronische Schaltkreise und ihre Anwendung, Berlin 1985

[4] *R. Egeler*, *ZX Spectrum* Hardware, Haar 1985

[5] *L. Schüssler*, *ZX Spectrum* Hardware-Erweiterungen, Düsseldorf 1984

[6] *M. James*, Der Weg zur *ZX-Spectrum*-Meisterschaft, Würzburg 1985

[7] *K. Schlenzig/S. Schlenzig*, Tips und Tricks für kleine Computer, Berlin 1988

[8] *A. Bogatz*, Mikrorechner in der Amateurmeßtechnik, Berlin 1988

[9] *M. Kramer*, Praktische Mikrocomputertechnik, Berlin 1987

[10] *H. Kieser/M. Bankel*, Einchipmikrorechner, Berlin 1986

[11] *ZX* Allerlei. In: Elektor 6/1984, S. 48ff.

[12] Joystick-Anschluß an den *ZX Spectrum*. *In:* *Radiotechnika* 4/1985, S. 7ff.

[13] Speichererweiterungen für den *ZX Spectrum*. In: Radiotechnika 7/1985, S. 9ff.

[14] *R. Arenz/M. Görlitz*, Das *Sinclair Spectrum* ROM, München 1984

[15] *Sinclair* Research Ltd., *ZX Spectrum* Benutzerhandbuch (Deutsche Übersetzung), München 1984

[16] *U. Fischer*, *Beta Basic* 3.0 Deutsches Handbuch, Hamburg 1985

Inhaltsverzeichnis

Computer werden heute im Heim-, Hobby- oder Geschäftsbereich vielseitig eingesetzt. Besonders der internationale Vertreter der 8-bit-Technik, der *ZX Spectrum,* ist auch in der DDR und im gesamten RGW-Bereich weit verbreitet. Zur Erhöhung der Leistungsfähigkeit dieses Heimcomputers werden in diesem Buch neben dem Aufbau u. a. folgende Hardware-Erweiterungen beschrieben:

* Erweiterungsanschluß des *ZX Spectrum*
* Reset-Taste
* Video-Anschluß
* Lichtstift
* serielle Schnittstellen
* Pegelanpassung für Kassettenrecorder
* 8-Kanal-Digitalvoltmeter
* Schaltkreistester
* Busverlängerung
* Adapter
* Speichererweiterungen

sowie

* Reparaturtips

ISBN 3-327-01030-7
DDR 6,20 M